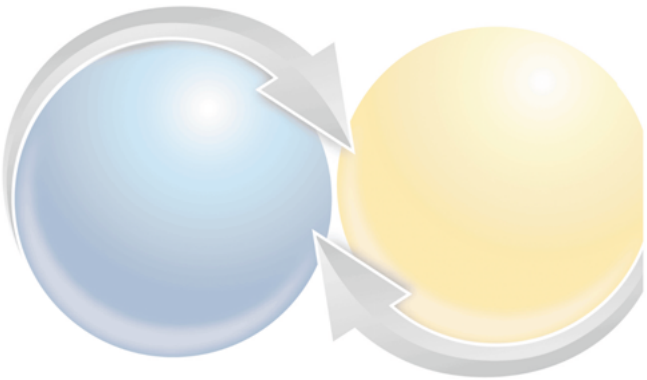
Livelink WCM Server

Content Client
User Manual

This manual describes how to use the Content client for editing contents of websites which are managed with Livelink WCM Server. It provides information on the following topics:

- introduction to handling and configuring the Content client
- adding, editing, and deleting objects
- staging and workflow in the WCM system
- working with InSite Editing and Content Assembly
- managing websites with the Content client



OPEN TEXT CORPORATION

Great Minds Working Together™

Copyright 2006 by Open Text Corporation.

The copyright to these materials and any accompanying software is owned, without reservation, by Open Text Corporation. These materials and any accompanying software may not be copied in whole or part without the express, written permission of Open Text Corporation.

Open Text Corporation is the owner of the trademarks Open Text, 'Great Minds Working Together', Livelink, and MeetingZone among others. This list is not exhaustive. All other products or company names are used for identification purposes only, and are trademarks of their respective owners. All rights reserved.

Open Text Corporation provides certain warranties and limitations in connection with the software that this document describes. For information about these warranties and limitations, refer to the license agreement entered into between the licensee and Open Text Corporation.

Contacting Us

Open Text Corporation
Corporate Headquarters
275 Frank Tompa Drive,
Waterloo, Ontario, Canada
N2L 0A1

(519) 888-7111

If you subscribe to our Customer Assistance Program or would like more information about the support program, write to Open Text Corporation's Customer Support at support@opentext.com or telephone (800) 540-7292 or (519) 888-9933. Our support hours are Monday through Friday, 8:30 a.m. to 8 p.m. (EST).

If you have comments or suggestions regarding this documentation, write to the Open Text Corporation Publications Group at documentation@opentext.com.

For more information about Open Text Corporation's products and services, visit our home page at http://www.opentext.com.

© 2006 IXOS SOFTWARE AG

Werner-v.-Siemens-Ring 20
85630 Grasbrunn, Germany

Tel.: +49 (89) 4629-0
Fax: +49 (89) 4629-1199
eMail: office@ixos.de
Internet: http://www.ixos.com

All rights reserved, including those regarding reproduction, copying or other use or communication of the contents of this document or parts thereof. No part of this publication may be reproduced, transmitted to third parties, processed using electronic retrieval systems, copied, distributed or used for public demonstration in any form without the written consent of IXOS SOFTWARE AG. We reserve the right to update or modify the contents. Any and all information that appears within illustrations of screenshots is

provided coincidentally to better demonstrate the functioning of the software. IXOS SOFTWARE AG hereby declares that this information reflects no statistics of nor has any validity for any existing company. This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit (http://www.openssl.org/) and software developed by the Apache Software Foundation (http://www.apache.org/).

Trademarks IXOS: IXOS SOFTWARE AG.

SAP®, R/3® and SAP ArchiveLink® are registered trademarks of SAP AG.

Microsoft®, Microsoft Windows NT® and the names of further Microsoft products are registered trademarks of Microsoft Corporation.

Acrobat Reader Copyright © 1987 Adobe Systems Incorporated. All rights reserved. Adobe and Acrobat are trademarks of Adobe Systems Incorporated which may be registered in certain jurisdictions.

Siebel® is a registered trademark by Siebel Systems, Inc.

Other product names are used only to identify the products and they may be registered trademarks of the relevant manufacturers.

Copyright © 2006 Gauss Interprise AG Hamburg, Gauss Interprise, Inc. Irvine, California. All rights reserved worldwide.

This document and the related software are property of Gauss Interprise AG or its suppliers and protected by copyright and other laws. They are distributed under licenses restricting their use, copying, distribution, and decompilation. Neither receipt nor possession of this document confers or transfers any right to reproduce or disclose any part of the contents hereof. No part of this document may be reproduced in any form by any means without prior written authorization of Gauss Interprise AG or Gauss Interprise, Inc.

Moreover, the regulations of the software license agreement apply to this documentation.

All brand names and trademarks mentioned are the property of their respective owners.

http://www.opentext.com/bridging/gauss.html

Program version: Livelink Web Content Management Server™ (Content Server) 9.5.1

Document version: En-01

Publication date: May 2006

Table of Contents

List of Figures **8**

List of Tables **13**

Chapter 1 Introduction **17**

What Livelink WCM Server Has to Offer 17

About this Manual 21

Typographic Conventions 23

Chapter 2 Concepts **27**

System Architecture 27

Structure of a WCM Object 38

Functional Areas 61

Object Types 65

Chapter 3 Working with the Content Client **73**

Starting the Content Client 74

The User Interface of the Content Client 81

Editing Objects in the Content Client 86

The Log 103

Help on the Content client 105

Chapter 4 Configuring the Content Client **107**

Adding a Profile 109

Using a Profile 127

Editing a Profile 128

Deleting a Profile 128

Customizing the Content Client 129

Chapter 5 Adding Objects 133
Adding Objects on the Basis of an Object Type 134
Adding Objects on the Basis of a Template 148
Importing Linked Objects 154
Importing Objects from a ZIP File 158
Copying an Object 162
Chapter 6 Editing Objects 167
Checking Objects Out and In 168
Editing Objects 177
Editing Metadata 193
Maintaining References 201
Editing Access Rights 215
Deleting Objects 227
Converting Objects 233
Chapter 7 Submitting, Releasing, and Rejecting Objects 247
Submitting an Object 247
Releasing or Rejecting an Object 251
Chapter 8 Content Workflow 257
Structure of a Workflow Definition 258
Installing the Workflow Modeler 261
Working with the Workflow Modeler 262
Creating and Editing Workflow Definitions 272
WCM Objects in the Workflow 281

Chapter 9	**InSite Editing**	**293**
	Working with InSite Editing	294
	Application Example: The HeroLeasing Website	304
	Configuring InSite Editing	309
Chapter 10	**Working with Assembled Objects**	**317**
	General Procedure	318
	Editing Assembled Objects	324
	Application Examples	324
Chapter 11	**Livelink Contents in the Content Client**	**339**
	Adding Relators	340
	Editing Relators	353
	Deleting a Relator	362
Chapter 12	**Finding Objects and Editing Them Together**	**365**
	The Standard Filters	366
	The Filter Editor	376
Chapter 13	**Working with Templates**	**397**
	Adding Templates	399
	Editing and Deleting Templates	400
	Cascading Templates	401
Chapter 14	**Working with Special Attributes**	**409**
	Special Attributes in the Metadata Dialog Box	411
	Special Attributes in the Object Filter	412
	Statification	413
	Activating Thumbnails for Objects	422

Chapter 15 Working with WCM Tags 427
Integrating Metadata 428
Parameterized WCM Tags 440
Chapter 16 Managing Websites in the Content Client 465
Managing Object Types 466
Managing Attribute Sets 477
Managing Object Categories 485
Appendix A Livelink WCM Server and XML 493
The XML Object Types 493
Using Templates 498
Application Examples 500
Glossary 507
Index 515

List of Figures

Fig. 1 –	The staging of Livelink WCM Server	30
Fig. 2 –	Example workflow in Livelink WCM Server	31
Fig. 3 –	Object statuses in the Edit view	33
Fig. 4 –	Object statuses in the QA view	34
Fig. 5 –	Structure of a WCM Object	39
Fig. 6 –	Separation of layout and content	41
Fig. 7 –	References of a WCM object	48
Fig. 8 –	Absolute and relative references	49
Fig. 9 –	Assignment of users to groups and roles	53
Fig. 10 –	Inheritance of access rights	55
Fig. 11 –	Interrupted chain of inheritance as a result of changed access rights	56
Fig. 12 –	Restoring the inherited access rights	57
Fig. 13 –	Interpreting group- and role-specific access rights	59
Fig. 14 –	Interpreting group-, role-, and user-specific access rights	60
Fig. 15 –	Logging in to the Content client	75
Fig. 16 –	Selecting a website	76
Fig. 17 –	Change password	78
Fig. 18 –	Taking over as a substitute	79
Fig. 19 –	The Login info dialog box	80
Fig. 20 –	The user interface of the Content client	81
Fig. 21 –	The “Topic structure” view	83
Fig. 22 –	The “Template structure” view	83
Fig. 23 –	The “Object list” view	84
Fig. 24 –	The Favorites list	85
Fig. 25 –	Object information in the status bar	86
Fig. 26 –	Sending an e-mail	94

Fig. 27 – The “Object filter” view 97
Fig. 28 – The “My objects” view 98
Fig. 29 – Filters in the “My worklist” view 99
Fig. 30 – The “My worklist” view 99
Fig. 31 – The “Search” view 100
Fig. 32 – Result of a search 100
Fig. 33 – The action list 102
Fig. 34 – The Log dialog box 103
Fig. 35 – “Display version” function in the Log dialog box 104
Fig. 36 – The Configuration dialog box 108
Fig. 37 – Configuring the basic settings of a profile 110
Fig. 38 – Security warning when starting the integrated HTML editor 115
Fig. 39 – Configuring the toolbar 116
Fig. 40 – Configuring the action list 119
Fig. 41 – Adding an application class 122
Fig. 42 – Entering the file extension for an application class 123
Fig. 43 – Configuring the dynamic deployment 125
Fig. 44 – Specifying title, type, and workflow of a new object 136
Fig. 45 – Specifying the basic properties of a new object 138
Fig. 46 – Specifying the responsibilities for a new object 140
Fig. 47 – Specifying the attribute values for a new object 143
Fig. 48 – Specifying the property values for a new object 144
Fig. 49 – Selecting a workflow transition 145
Fig. 50 – Entering a remark for a new object 146
Fig. 51 – Selecting embedded objects when adding an object 147
Fig. 52 – Specifying title and template when adding objects on the basis of a template 150
Fig. 53 – Template selection with thumbnails 151
Fig. 54 – Specifying type and basic properties when adding objects on the basis of a template 152

Fig. 55 – Specifying title, file, and object type when importing linked objects 156
Fig. 56 – Specifying a ZIP file for import 159
Fig. 57 – Specifying title and type when importing from a ZIP file 161
Fig. 58 – Security warning when working with the Download applet for the first time 170
Fig. 59 – Saving a file manually 171
Fig. 60 – The integrated HTML editor 172
Fig. 61 – Selecting embedded objects when checking in an object 175
Fig. 62 – Selecting a WCM tag 180
Fig. 63 – Specifying parameters for a WCM tag 182
Fig. 64 – Specifying additional parameters for a WCM tag 183
Fig. 65 – Dialog box for replacing an object 191
Fig. 66 – Selecting embedded objects when replacing an object 192
Fig. 67 – The Metadata dialog box 194
Fig. 68 – The References dialog box 202
Fig. 69 – Topic structure when adding references 205
Fig. 70 – Topic structure when adding references 207
Fig. 71 – The Access rights dialog box 217
Fig. 72 – Adding a principal to the access control list 218
Fig. 73 – Editing access rights 221
Fig. 74 – Converting a Word file into HTML 234
Fig. 75 – Converted objects in the navigation area 238
Fig. 76 – The compound object in the navigation area 244
Fig. 77 – Selecting dependent objects when submitting an object 249
Fig. 78 – Selecting dependent objects when releasing an object 252
Fig. 79 – Selecting dependent objects when releasing an object directly 254
Fig. 80 – Security warning when starting the workflow modeler 264
Fig. 81 – Dialog box when adding an activity 274
Fig. 82 – Dialog box for assigning a workflow 283

Fig. 83 – Dialog box for selecting a transition 284
Fig. 84 – Example workflow for editing data sheets 288
Fig. 85 – Logging in to InSite Editing 295
Fig. 86 – InSite Editing 296
Fig. 87 – Editing an object fragment via InSite Editing 301
Fig. 88 – Replacing the referenced object via InSite Editing 303
Fig. 89 – The template “Herotemplate” in the “Content (WYSIWYG)” view 305
Fig. 90 – Teasers on the website “HeroLeasing” 307
Fig. 91 – Replacing a teaser in the template “Herotemplate” 309
Fig. 92 – Configuring the `VipHCLApplication` 313
Fig. 93 – Structure of the “PressRelease” template 322
Fig. 94 – The *Properties* tab in the Metadata dialog box 323
Fig. 95 – The template “Partnertemplate” in the object preview 329
Fig. 96 – Entering partner information for the assembled object during creation 330
Fig. 97 – Partner profile in the object preview 331
Fig. 98 – Partner object in InSite Editing 334
Fig. 99 – Editing a partner object via InSite Editing 335
Fig. 100 – Changing the partner logo via InSite Editing 336
Fig. 101 – Specifying the Livelink system 342
Fig. 102 – Selecting the Livelink object when adding a relator 343
Fig. 103 – Specifying the version of the Livelink object 344
Fig. 104 – Entering comments about the relator 346
Fig. 105 – Selecting the Livelink folder 347
Fig. 106 – Specifying the version of the Livelink folder relator 348
Fig. 107 – Specifying title, file, and object type 351
Fig. 108 – The Livelink metadata dialog box 356
Fig. 109 – The category “training materials” in the Livelink system 360
Fig. 110 – Standard filters of the object filter 367

Fig. 111 – Standard filters of the “My objects” view 367
Fig. 112 – Filter hitlist 371
Fig. 113 – The filter editor 376
Fig. 114 – Specifying the Filter Criterion 378
Fig. 115 – Determining a comparator 379
Fig. 116 – Specifying a comparison value 380
Fig. 117 – Generated filter expression 381
Fig. 118 – Adopting the filter expression 381
Fig. 119 – Saving a user-defined filter 382
Fig. 120 – Entering the value for the filter variable 383
Fig. 121 – Searching with a new search expression 383
Fig. 122 – Defining a logical operator 384
Fig. 123 – Example of a cascading template 402
Fig. 124 – Merging the body element in a cascading template 404
Fig. 125 – Deleting a template from a cascade 407
Fig. 126 – The object category “Invoice” in the Metadata dialog box 411
Fig. 127 – Filtering for the payment deadline Nov 30, 2004 413
Fig. 128 – Selecting an image for a thumbnail 424
Fig. 129 – Selection dialog box for the VIPKEYWORDLIST attribute 457
Fig. 130 – Overview of available object types 466
Fig. 131 – Specifying general information for the object type 468
Fig. 132 – Specifying object data type and additional settings for the object type 470
Fig. 133 – Making assignments for the object type 474
Fig. 134 – Default attribute sets 478
Fig. 135 – Dialog box when creating an attribute 481
Fig. 136 – Dialog box when creating a property 488
Fig. 137 – Result of the transformation on the client side 502
Fig. 138 – Result of the transformation on the server side 505

List of Tables

Table 1 –	Object statuses with description	35
Table 2 –	Object types with possible content components	40
Table 3 –	Standard metadata with description	43
Table 4 –	Actions and required access rights	50
Table 5 –	Example of group-roles and rights	54
Table 6 –	Functional areas and object types	62
Table 7 –	Object types with description	66
Table 8 –	Staging steps and e-mail notification	93
Table 9 –	Activating links for e-mails in the **mail-conf.xml** file	95
Table 10 –	Basic settings of a profile	111
Table 11 –	Navigation settings of a profile	118
Table 12 –	Settings for the object attributes in a profile	121
Table 13 –	Expressions for identifying references	211
Table 14 –	Object statuses and possible initial activity	259
Table 15 –	Possible transitions in the workflow and required access rights	260
Table 16 –	Commands in the menu bar of the workflow modeler	265
Table 17 –	Graphical representation of workflow elements	271
Table 18 –	The group “Marketing”	287
Table 19 –	Configuration items of InSite Editing	310
Table 20 –	Important data types with formats used	319
Table 21 –	Properties of the object category “PressRelease”	321
Table 22 –	Properties of the object category “Partner”	326
Table 23 –	WCM tags for the most common Livelink metadata	358
Table 24 –	Standard filters in the object filter	368
Table 25 –	Standard filters in the “My objects” view	369
Table 26 –	Comparators in the filter editor	386

Table 27 – Comparison values in the filter editor 388
Table 28 – Examples of the spelling of comparators 389
Table 29 – Combination operators in the filter editor 390
Table 30 – Principal operators in the filter editor 391
Table 31 – Examples for Filter Expressions 393
Table 32 – Body tag when combining object and template 398
Table 33 – Sample configuration for the special attribute “Statify page” 417
Table 34 – Sample configuration for the special attribute “Absolute URL” 420
Table 35 – WCM tags for object-specific information 429
Table 36 – WCM tags for global information 432
Table 37 – WCM tags for workflow information 433
Table 38 – WCM tags for the supplied special attributes 433
Table 39 – Data types with formats used 436
Table 40 – Values of the `format` parameter for the data types “Date”, “Time”, and “Time stamp” 442
Table 41 – Examples of the `format` parameter for the data type “Time stamp”. 444
Table 42 – Examples of the `format` parameter for the data type “Object URL” 445
Table 43 – Values of the `format` parameter for the data type “Floating-point number”. 445
Table 44 – Examples of the `format` parameter for the data type “Floating-point number” 446
Table 45 – Examples for using the `type` parameter 451
Table 46 – Data types and `target` parameter 462
Table 47 – Object data types and WCM objects 471
Table 48 – MIME types and WCM objects 472
Table 49 – Content and file extension depending on the template 498

CHAPTER 1
Introduction

This chapter gives you an introduction to the functionality of Livelink Web Content Management Server™ (Livelink WCM Server for short). It also provides a brief overview of the structure of this manual.

What Livelink WCM Server Has to Offer

Many companies today have a number of intranet, extranet and Internet environments with specific focuses. This involves managing content for a wide variety of target groups: employees, subsidiaries, business associates, or customers. An efficient content management system in the company makes it possible to equip users with the rights and tools they need to provide such content, and at the same time ensure control of the content.

Livelink WCM Server enables you to implement sophisticated intranet, Internet, and extranet solutions. The content is managed in the form of complex websites. Livelink WCM Server includes other products for creating and managing dynamic and personalized websites.

Easy to Use

Livelink WCM Server is designed as a client-server solution. All website data is managed on servers. Website content can be edited with the Content client, which does not require separate installation on the client computers. The Content client offers editors, graphics specialists, or quality assurance employees an intuitive user interface for editing website content. Any desired applications can be used for editing content, which means that authors can continue to use their familiar authoring environment and concentrate on their specific subject areas.

The elements of the website are shown in a clearly arranged tree structure. An integrated HTML editor with design view allows you to edit HTML documents even without separate applications.

The Content client allows authors to edit website content no matter where they may be – access to the server on which a website is hosted is possible via the LAN, WAN, or Internet. Thanks to its extensive access control functions and communication encryption, Livelink WCM Server guarantees protection for sensitive data.

Defined Creation and Approval Process

All objects of the websites that you manage with Livelink WCM Server pass through fixed stages: editing, quality assurance, and publication for productive operation. Every editor must submit new or modified objects to Quality Assurance. Here, the object is checked and either released for publication in the intranet or Internet, or rejected and returned to the responsible editor for correction.

This structured creation and approval process helps you ensure compliance with quality standards.

Simultaneous Editing by Different Users

With Livelink WCM Server, several users can edit website objects at the same time. Whether a user can edit a specific object depends in particular on the current status of the object and on the user's access rights.

If an editor checks out an object for editing, this object is locked for write access by other users. For each object, there is a log telling you, among other things, which user has checked out the object. When the object is checked in again, the changes become visible to other users. This mechanism guarantees structured work procedures for editing website objects and thereby ensures consistency of the company's data.

Separation of layout and content

Thanks to its separation of layout and content, Livelink WCM Server helps you implement corporate design guidelines for a consistent company-wide layout. Each page in a website can be assigned a defined template, which can, however, be changed at any time. This template contains the layout information for the page. Changes to the template result in fully automatic changes in the layout of all pages concerned. In other words, you change the layout in one place only – the template – and your entire website is automatically updated by Livelink WCM Server.

The template is a freely designable page with placeholders for information from the object's metadata and for the actual content. During page generation, Livelink WCM Server automatically merges the template and the content.

Up-to-Date Content

Up-to-date information is the key to customers' purchase decisions, and thus an essential requirement for enterprise portals. To ensure that website content is up-to-date, Livelink WCM Server offers a number of functions:

- content management during the entire life cycle and throughout all phases of the staging, which consists of editing (Edit view), quality assurance (QA view), and publishing (Production view)
- support of time-controlled publication ("delayed release")
- placeholders for object expiration dates
- full versioning of content
- integrated reference check to avoid "dead links" or graphics that cannot be displayed

Integration with Livelink Enterprise Server

Due to the integration of Livelink WCM Server with Livelink Enterprise Server, you can add Livelink objects as WCM objects to a WCM-managed website. These WCM objects are called *relators*. They always refer to a certain version of the Livelink object. The WCM object type "Livelink relator" enables you to integrate a single Livelink object in a website; the WCM object type "Livelink folder relator" is provided for integrating Livelink folders.

About this Manual

This manual describes the functions of Livelink WCM Server from the point of view of a user who is actively involved in editing or quality assurance of websites with Livelink WCM Server. In addition to this User Manual, the following sources provide information:

- Livelink Integration Manual: This document explains how to integrate your WCM system with a Livelink system.
- Livelink WCM Server Administrator Manual: This document explains how to configure and administer WCM systems. Moreover, it contains a detailed description of the technical concepts of Livelink WCM Server.
- Livelink WCM Server Installation Manual: This document describes the installation of the WCM system and provides information on configuring HTTP server and JSP engine.
- WCM Java API Programmer's Manual: This document contains information on interfaces, classes, and methods of the Java programming interface (WCM Java API), which enables you to use the functionality of the WCM servers via external programs.
- Online help: Livelink WCM Server offers an integrated online help that can be called at any time when working with the Content client.

The contents of this manual are organized as follows:

- Chapter 2 “Concepts” explains the fundamental concepts of Livelink WCM Server.
- Chapter 3 “Working with the Content Client” describes the first steps in working with Livelink WCM Server and introduces the user interface of the Content client.
- Chapter 4 “Configuring the Content Client” explains how to tailor the process of editing objects and the layout of the Content client to your needs.
- Chapter 5 “Adding Objects” offers a detailed description of the different procedures for creating new WCM objects.
- Chapter 6 “Editing Objects” gives you a detailed introduction to object editing.
- Chapter 7 “Submitting, Releasing, and Rejecting Objects” describes the respective staging actions.
- Chapter 8 “Content Workflow” explains how to create user-defined workflows in order to extend the staging of Livelink WCM Server.
- Chapter 9 “InSite Editing” shows how objects can be edited directly in the website.
- Chapter 10 “Working with Assembled Objects” introduces the functions of Content Assembly and explains how to edit assembled objects.
- Chapter 12 “Finding Objects and Editing Them Together” shows how to use the search functions of the Content client.
- Chapter 13 “Working with Templates” describes the special features of WCM templates.

- Chapter 14 “Working with Special Attributes” explains the special attributes.
- Chapter 15 “Working with WCM Tags” shows how to use WCM tags for integrating object information into web pages.
- Chapter 16 “Managing Websites in the Content Client” introduces the functions the Content client offers for managing websites.
- Appendix A “Livelink WCM Server and XML” describes how to use the four XML object types. For this chapter, a sound knowledge of XML and Livelink WCM Server is required.

Typographic Conventions

The following conventions are used in the text to draw attention to program elements, etc.:

Element	Font or symbol	Examples
Program interface such as menu commands, windows, dialog boxes, field and button names	*Menu* → *Entry*	*File* → *Create*
Paths to directories, file and directory names	**Drive:\Directory\File name**	**D:\WCM\ admin.bat**

Element	Font or symbol	Examples

Quotations from program code or configuration files	`Code quotations`	`<head>`
		`<title>heading </title>`
		`</head>`

Variables, i.e. placeholders for specific elements	{variable}	{WCM installation directory}

Important **information** and **warnings** are enclosed in gray boxes. Make sure to read such information to avoid losing data or making errors when using and managing WCM systems.

CHAPTER 2
Concepts

This chapter gives you a basic knowledge of the architecture of a Web Content Management System (WCM system) and of the way Livelink WCM Server works. This knowledge is required for using Livelink WCM Server.

System Architecture

Livelink WCM Server is based on client-server technology. This technology refers to a system architecture in which the processing of an application is split into two separate parts. One part runs on the server (high-performance computer or process), the other on a client (workstation or program). The two parts are joined by networks to form a system. The client sends a request to the server to process data and uses the services of the server. Applied to Livelink WCM Server, this means that management of the website data and users takes place on the server. By contrast, editing of data by editors or graphics specialists is done via the Content client.

In a WCM system, one or more *websites* can be managed. A website can be published for Internet, intranet, or extranet access. The website objects that you manage with Livelink WCM Server (HTML pages, images, JSP scripts, etc.) are referred to as *WCM objects*. They are stored in a database.

Website objects in Livelink WCM Server go through clearly defined stages of editing, quality assurance, and publication for productive operation. This process is called *staging* and applies to all objects managed with Livelink WCM Server. Depending on the processing status, there are different views of the WCM objects. For example, there may be changes to an object that have not yet been submitted to QA, and of course, not yet released for productive operation.

The different views of the WCM objects are provided by the so-called *deployment systems* which are managed by means of the Admin client. Deployment systems generate pages for the WCM objects stored in the database. These pages can be displayed in a browser.

In addition to the staging, Livelink WCM Server enables you to assign a workflow to the objects in order to define several steps of editing and quality assurance.

The Staging and Workflow of Livelink WCM Server

All objects of the websites that you manage with Livelink WCM Server pass through fixed stages: editing, quality assurance, and publication for productive operation. Editing and quality assurance of the objects is carried out in the Content client. Releasing publishes the objects in the website. Passing through these stages is called *staging*.

In addition to the staging, you can assign a *workflow* to certain objects, e.g. for successive editing by several editors or for multi-stage quality assurance.

Staging

According to the processing status of a WCM object, there are different views of the object: the Edit view, the QA view, and the Production view.

Edit view

In the Edit view, WCM objects, such as HTML pages, are added and edited by editors and graphics specialists. These users need corresponding access rights for the objects. After they have been edited, the objects are submitted to Quality Assurance. This makes the changes to the objects visible in the QA view.

QA view

The QA view shows the WCM objects with all changes that have been submitted to Quality Assurance. Quality Assurance staff can check the changes with regard to content and formal aspects. After checking an object, they decide whether it is to be returned or released. For release, the respective access right is required. The release transmits the quality-assured version of the object to the Production view, and thus makes it available to the end user in the published website.

Production view

The Production view makes the released pages of a website available to the website visitors. By means of a web server, these pages can be accessed in the Internet, intranet, or extranet.

The following figure illustrates the different staging steps and the actions that realize the transitions between these steps.

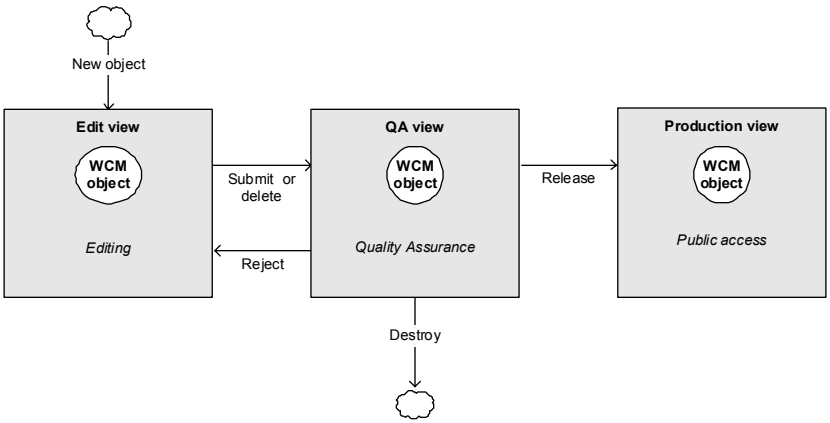


Fig. 1 – The staging of Livelink WCM Server

Workflow

To extend the predefined editing and publication stages of WCM objects by custom workflow steps, you can assign a workflow to a WCM object. Depending on the processes in your company, this workflow may contain several Edit and QA steps.

When the workflow is created in the workflow modeler, certain users are assigned to the workflow steps. Only these users are allowed to edit the respective objects within these workflow steps. Of course, the users also need sufficient object rights.

After editing the WCM object, the user simply selects one of the predefined workflow steps in order to forward the object to the next user or to return it to the previous workflow step. The “normal” staging actions (submit, reject, release, etc.) are not available. The necessary staging transitions are automatically performed in the background.

The following diagram illustrates an example workflow.

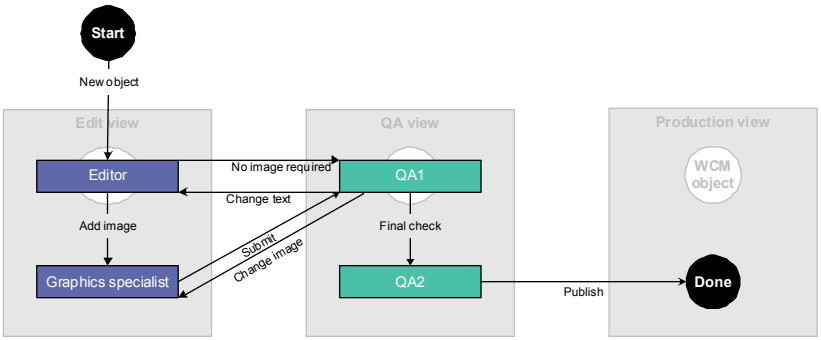


Fig. 2 – Example workflow in Livelink WCM Server

The following sections provide information on different aspects of the staging and workflow of Livelink WCM Server:

- status changes of a WCM object in the staging and workflow, see the following section
- tasks of an editor, see section “Tasks of an Editor” starting on page 37
- tasks of quality assurance, see section “Tasks of a QA Staff Member” starting on page 37

Statuses of WCM Objects

As a result of the various work steps, the status of an object changes when it is edited. The current object status determines the functions you can perform on the object. In the Content client, different icons illustrate the object statuses.

You can inform the person responsible for the next work step by e-mail when the status of an object changes. For example, if you have edited an object and checked it in again, the object acquires the status “changed” and can be submitted to Quality Assurance. When you submit the object, you can inform the responsible staff by e-mail. The required e-mail addresses are stored in the object's metadata.

If a workflow is assigned to the object, the e-mail is automatically sent to the address of the user who is assigned to the next workflow step.

Example

For a better understanding, the staging is described on the basis of the following example:

1. You add an object. The new object's status is “changed”. This status is indicated by the icon (two arrows).
2. You check out the object in order to edit it. The respective file is copied to the local working directory you configured and gets the status “checked out”. This means that no other user can edit the object until you check it in. The relevant icon is .

3. After editing the object, you check it. The object again has the status “changed” .

   You can repeat steps 2 and 3 as many times as you like.

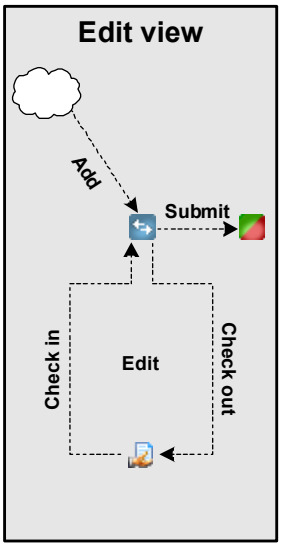


Fig. 3 – Object statuses in the Edit view

4. You submit the object to Quality Assurance. The object thus has the status “submitted”. The relevant icon is (red-and-green split square). The changes made in the Edit view now become visible in the QA view. An object with the status “submitted” cannot be checked out for editing.

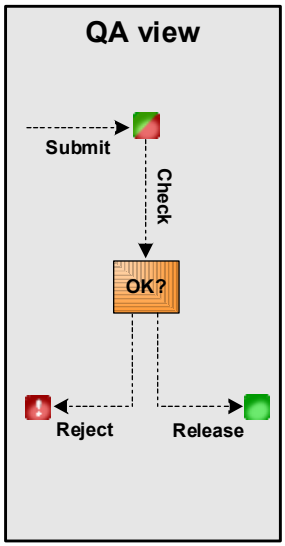


Fig. 4 – Object statuses in the QA view

5. The object is checked by Quality Assurance. Quality Assurance can either release or reject the object. If Quality Assurance rejects the object, it is given the status “rejected” and returned for correction. The relevant icon is (red square with exclamation mark). You can then check out the object again and edit it. Afterwards, you submit the object to Quality Assurance again.
6. If the object is released by Quality Assurance, it acquires the status “released”. The relevant icon is (green square). After release, the new version of the object is available to the end users in the Production view of the website.

Both in the staging and in the workflow of Livelink WCM Server, the object always has a defined status. This status specifies which actions you can perform on the object. For example, only submitted objects can be released by Quality Assurance. The function “Release” cannot be applied to other objects.

If a workflow is assigned to the object, the functions you can perform on the object also depend on the current workflow step of the object. For example, if the workflow defines a two-stage quality assurance and the object is in the first stage of quality assurance, it cannot be released despite having the status “submitted”. It can only be released in the second stage of quality assurance.

The following table provides an overview of the main features of the various object statuses.

Table 1 – Object statuses with description

Object status	Staging icon	Workflow icon	Description
changed			The object has been changed. New objects also have this status.
checked out			The object has been checked out for editing and is thus barred for write access by other users.
delayed release			The object has been checked and released by Quality Assurance. However, the release date specified for the object has not yet been reached. The current version is thus not yet available to the end users in the Production view.

Object status	Staging icon	Workflow icon	Description
deleted			The object has been deleted. In the case of objects that have already been released, this process must be confirmed by Quality Assurance.
rejected			The object has been checked and rejected by Quality Assurance. It can be checked out for further editing and changed. Afterwards, the object must be submitted to Quality Assurance again.
released			The object has been checked and released by Quality Assurance. The current version is thus available to the end users in the website's Production view. The object can be checked out and edited again. **Note:** After an object has passed through a workflow and has been released, the assignment between the object and the workflow is removed. The object gets the status "released" (staging icon). If a workflow is assigned to a released object, this is marked by the respective workflow icon.
submitted			The object has been submitted to Quality Assurance and its current version is available in the QA view. An object with this status cannot be checked out for editing.

Tasks of an Editor

The editors are responsible for creating and maintaining the website content. They have to publish information on an up-to-the-minute basis.

The right to add and edit objects is reserved to certain groups, roles, or individual users – the so-called *principals*. Only principals that are assigned to certain functional areas and have the access right “Create”/ “Change object” are allowed to add and edit objects. Assigning access rights enables you to define clear responsibilities, which reflect the organizational structure of your company.

For further information on defining groups and roles and on assigning access rights, refer to section “Access Rights” starting on page 50.

Tasks of a QA Staff Member

Correct and thorough quality assurance for all objects in a website is an important precondition for successful use of the website. Thus, the staging concept of Livelink WCM Server takes account of quality assurance as a part of content management. This guarantees a better quality of the website content.

In addition to the designers and editors, who are responsible for layout and content of the website, there is staff whose task is the quality assurance of the website. Whether employees are allowed to perform quality assurance tasks depends on their rights. For example, to release objects and thus publish them in the Production view, the access right “Release” is required.

The task range of the QA staff includes:

- monitoring the quality standards defined by the company
- checking new or modified objects
- monitoring content and layout of the published objects

- checking for up-to-dateness and taking appropriate action if the expiration date of an object is reached
- checking the reference structure of the website
- monitoring the published pages for consistency of content
- compiling a list of the necessary changes when objects are rejected

Structure of a WCM Object

Each WCM object consists of certain components representing the different data of the object:

- content (optional), see the following section
- template (optional), see section “Template” starting on page 41
- metadata, see section “Metadata” starting on page 42
- references (optional), see section “References” starting on page 46
- access rights, see section “Access Rights” starting on page 50

All data of a WCM object is stored in an external relational database.

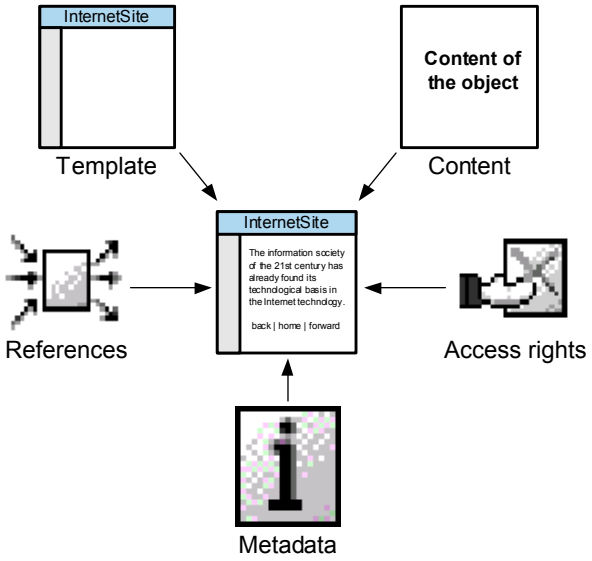


Fig. 5 – Structure of a WCM Object

Content

The content is the most important component of a WCM object. For an HTML page, it is the body element of the HTML page; for a graphic, it is the image file; for a frame object, the frameset, etc. The following table shows the various kinds of content components.

Table 2 – Object types with possible content components

Object type	Content component
ASP page, ASP topic, HTML page, JSP page, JSP topic, PHP page, PHP topic, Topic	The body element of the HTML file (If no template is assigned to the object, the content component corresponds to the complete source file.)
ASP template, HTML template, JSP template, PHP template	The head and body elements of the HTML file
Frame, Frame topic	The head and frameset elements of the source file
GIF image, JPEG image, PNG image	The complete graphics file
CGI script, Compound object, Excel document, Form instance, Form template, JavaScript, Other, PDF document, PowerPoint file, Word document	The complete source file
XML document, XML template, XSLT document, XSLT template	The content of an XML or XSLT document begins with the tag that does not correspond to the form <! or <?.
Workflow	The workflow definition

If you have assigned a template to the object, the WCM system automatically merges the template and the object content. When the object is displayed, you see the resulting HTML page. However, when checking out an object in the Content client, you edit only the content.

Note: Strictly speaking, it is possible to create an object without any content. When you add an object without specifying a file, this object initially has no content. The first time the object is checked out and then checked in again, it is given an “empty” content component. Objects of type “Assembled object”, which never have a content component, are the only exception.

Template

Separation of layout and content is an important property of WCM objects. You can define the layout of a WCM object in a template. The template is a freely designable page. It contains design elements, such as graphics, background images, background colors, text and link colors, as well as font formats. Additionally, the page contains placeholders for the actual content of the objects based on this template. Moreover, the metadata associated with the object can be referenced in the template.

When adding an object in the Content client, you specify whether the object is to use a template or not. It is also possible to assign a template to the object later or to change the assignment. If you decide to use a template, its layout will automatically be applied to the new object. Layout and content are separated, as illustrated in the following diagram.

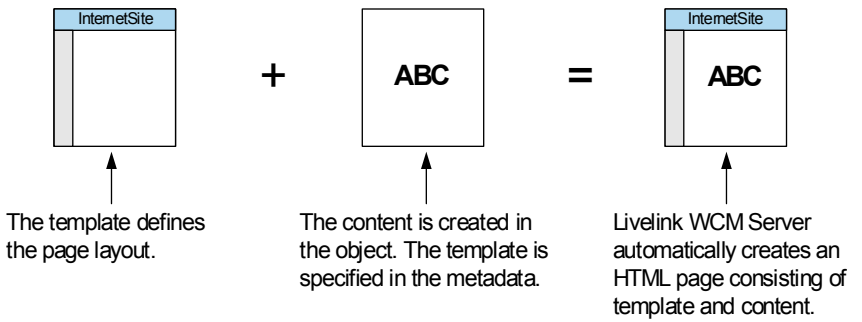

InternetSite
+
ABC
=
InternetSite
ABC
The template defines the page layout.
The content is created in the object. The template is specified in the metadata.
Livelink WCM Server automatically creates an HTML page consisting of template and content.

Fig. 6 – Separation of layout and content

If you use templates for the objects in your website, the pages in the website are generated by merging template and object content. The layout of the pages is defined in the templates. By means of placeholders – the so-called *WCM tags* – available in Livelink WCM Server, you can insert additional information from the object metadata in the generated HTML page. For example, by using the WCM tag {`VIPDATE_CREATED`}, you can display the object's creation date on the HTML page (see chapter 15 "Working with WCM Tags").

After a template has been changed, the changes are applied automatically to all objects based on that template as soon as the edited template is released. There is no need to adapt the individual objects. Like all WCM objects, templates are subject to the defined staging steps. This ensures that changes to a template do not become visible in the Production view until the template has been checked and released by Quality Assurance.

For detailed information on working with templates, refer to chapter 13 "Working with Templates".

Metadata

For each WCM object, Livelink WCM Server manages information describing the object. Such information is known as metadata, and comprises various properties of the object.

The metadata can be used to filter objects in Livelink WCM Server and process them accordingly. For example, you can use the object filter to display a list of all objects that have expired, or all objects that have been rejected by Quality Assurance.

By using WCM tags in the templates or in the HTML pages, it is possible to integrate information from the metadata, e.g. the object's expiration date, in the website pages (see chapter 15 "Working with WCM Tags").

Livelink WCM Server distinguishes two types of metadata: *standard metadata* and *special attributes*.

Standard Metadata

Standard metadata are metadata items that are defined by the WCM system and are present for each WCM object. The following table provides an overview of the standard metadata. The "Inherited" column indicates which metadata settings are adopted from the parent topic when the object is added.

Table 3 – Standard metadata with description

Metadata item	Description	Inherited
Author	The user who added the object	
Created on	The date (with time) on which the object was created	
Delayed release	The date (with time) at which the object version is to be published in the Production view	
Description	The description of the object content. This information can be used as a search criterion in the object filter.	
Direct release	Indicates whether the object can be released directly without having to go through Quality Assurance.	
E-mail Edit, E-mail QA, E-mail Release	The e-mail addresses defined for the object. Messages are sent to these addresses when the object acquires a certain status.	✓

Metadata item	Description	Inherited
Expiration date	The date (with time) until which the object remains valid. Expired objects can no longer be submitted to Quality Assurance.	✓
Heading	The heading of the object	
Keywords	The keywords describing the content of the object. This information can be used as a search criterion in the object filter.	
Language	The language of the object. This metadata item has no influence on the object content, but is used for classification purposes and as a search criterion in the object filter. Moreover, certain metadata, such as the date, are displayed in the selected language if they are integrated into the HTML page generated for the object by means of placeholders, so-called WCM tags.	
Modification date	The date (with time) on which the object was changed last	
Modified by	The ID of the last user who changed the object	
Object category	Indicates whether the object has been assigned to an object category (e.g. “Invoice”) and, if so, which object category is used	
Object size (in bytes)	The size of the object content	

Metadata item	Description	Inherited
OID	The **O**bject **Id**entification of the object (integer). The WCM system automatically generates an OID for every new object. You cannot change the OID. It is not reused after the object has been deleted. The OID is displayed in almost all dialog boxes. **Note:** Use the *OID* field in the toolbar to directly navigate to an object. If you frequently edit certain objects, it might be helpful to note down the respective OIDs.	
Path	The position of the object in the topic structure of the website	
Released by	The ID of the last user who released the object	
Release date	The date (with time) on which the object was released last	
Status	The current processing status of the object	
Suggested file name	The suggested name for the page generated from the WCM object. In many cases, the WCM system considers this suggestion.	
Target group	The group of people the object content is relevant to. This information can be used as a search criterion in the object filter.	✓

Metadata item	Description	Inherited
Template	Indicates whether a template is assigned to the object and, if so, which template is used **Note:** If you do not have read rights for the template assigned to the object, the entry “No read rights for template” is displayed in the Metadata dialog box. In this case, you cannot see the title of the template, only its OID is visible to you.	✓
Title	The name of the object	
Type	The object type of the object	
Version	The current three-digit version number of the object	

Special Attributes

Special attributes are all metadata that do not belong to the standard metadata. For detailed information on special attributes, refer to chapter 14 “Working with Special Attributes”.

References

The links between different objects are called references. You can only add references to objects that have a content component. References are used primarily for the following purposes:

- integrating the content of another file (e.g. a graphics file)
- linking to another file or HTML page

Links between different objects make it possible to establish branched reference structures. There are two different forms of references that can be used to integrate an object in the reference structure:

- references to the object; the object is the destination of the references
- references to other objects; the object is the source of the references

When editing an object, you can directly insert hyperlinks. These hyperlinks are automatically inserted as references in the metadata when the object is checked in. An object's references are modified accordingly if you move the object to a different location within the website.

All references that refer to resources outside the current website (e.g. to a page in the WWW or a file in the file system) are known as *external references*. References to objects in another website managed with Livelink WCM Server are also regarded as external references.

Note: When objects are changed, external references are not adjusted automatically by Livelink WCM Server, i.e. they are not subject to the reference check.

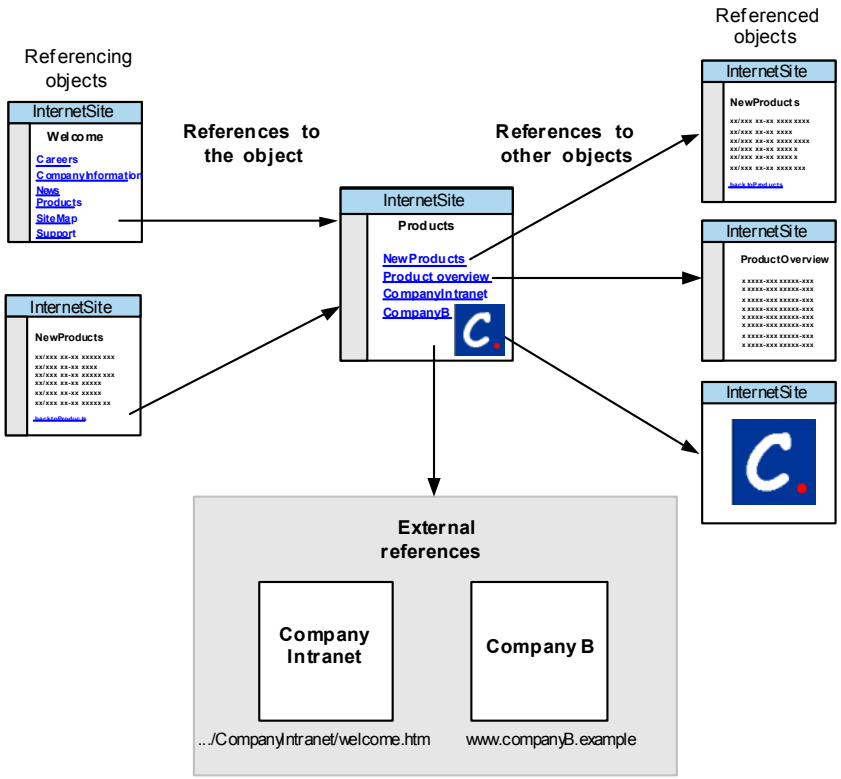


Fig. 7 – References of a WCM object

Absolute and Relative References

When creating hyperlinks in an HTML page, a distinction is made between *absolute* and *relative* references. An absolute reference contains the complete address of the destination object. A relative reference, by contrast, only specifies that part of the address that differs from the address of the source page.

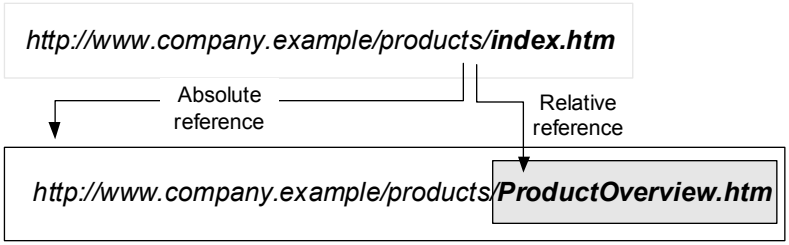


Fig. 8 – Absolute and relative references

Assume you have created an HTML page with the name **index.htm** and want to insert a reference to another page within the same directory, e.g. **ProductOverview.htm**. For the address of the referenced page, you can specify the complete address (absolute reference):

```
http://www.company.example/products/ProductOverview.htm
```

or you can specify the file name without path information (relative reference), if the file is located in the same directory as the referencing page:

```
ProductOverview.htm
```

By means of relative references, you can also address files in subdirectories. Assume that you have a page **NewPriceList.htm** that is located in the subdirectory **/prices/**. To enter a reference from the page **ProductOverview.htm** to this page, enter the relative sub-path:

```
/prices/NewPriceList.htm
```

Access Rights

The object properties define which users, groups, or roles are allowed to access an object. You can set the access rights separately for each object. In this way, it is not only possible to generally allow or forbid access, you can also set separate access rights for the individual users, groups, and roles. This function of Livelink WCM Server is known as *access control*. Every WCM object has an *access control list*, which specifies the users, groups, and roles that have access to that object.

Actions and Access Rights

The access rights determine the way in which users can access the object, e.g. whether they may only read the object or can edit it as well. In the settings of an object, each access right can be explicitly assigned or denied to a principal, i.e. a user, group, or role.

The following table gives an overview of the actions you can perform on objects and the required access rights.

Table 4 – Actions and required access rights

Action	Required access rights
Add object	Read + Create + Change object (for the parent topic)
Delete object	Read + Delete + Change object
Submit object	Read + Change object
Edit an object (check out, change, check in)	Read + Change object
Change an object's metadata	Read + Change metadata
Change an object's references	Read + Change metadata
Rename object	Read + Change metadata

Action	Required access rights
Move object	Read + Move and copy + Change metadata (for the source object)
Copy object	Read + Move and copy + Change metadata (for the source object)
Insert object (after cutting or copying)	Read + Create (for the target topic)
Check access rights of object	Read
Change access rights of object	Read + Change rights + Change metadata
Release or reject object	Read + Release
Release object directly	Read + Change object
Destroy object	Read + Release

Users, Groups, and Roles

Livelink WCM Server allows you to assign the users of a website to groups and to control access by assigning rights to these groups. If necessary, specific access rights can be given to individual users – this should, however, be the exception.

A role – like a group – is also a collection of users. Roles, however, are defined in terms of tasks, whereas groups are generally tied to organizational structures, such as departments or projects. Depending on the tasks, the roles in an organization may be defined in different ways. Some typical user roles that may be involved in editing a website are editors, graphics specialists, project managers, and QA staff.

Specific access rights for objects can be defined for a role – just as they can for a group. Since roles are usually defined on the basis of tasks, it may be even easier to assign access rights to a role than to a group, since editors, for example, need specific access rights, such as “Create” and “Change object”. Groups, by contrast, are often based on organizational units, e.g. sales, marketing, production, or administration.

The principal “World”

In Livelink WCM Server, there is the special principal “World”. This principal is a group and represents all users of the WCM system. Thus, the access settings for this principal are valid for all logged-in users. The access rights of the group “World” are evaluated on the basis of the usual priority rules (see section “Priority Rules” on page 57).

Group-Roles

You can allow access to an object for a group, a role, or a combination of a specific group and a specific role. Such a combination is called *group-role*. The group-role is not an organizational unit that is defined in the Admin client, but a configuration option in the Content client. The following example illustrates the concept of group-roles.

Example

Several groups and the roles “Editor” and “QA” were defined in the Admin client. The users were assigned to the groups on the basis of their departments. The users in the marketing department belong to the group “Marketing”, those in the training department belong to the group “Training”, etc. The users in each department who are responsible for adding and editing objects are assigned to the role “Editor”, while the users responsible for quality assurance of the objects are assigned to the role “QA”.

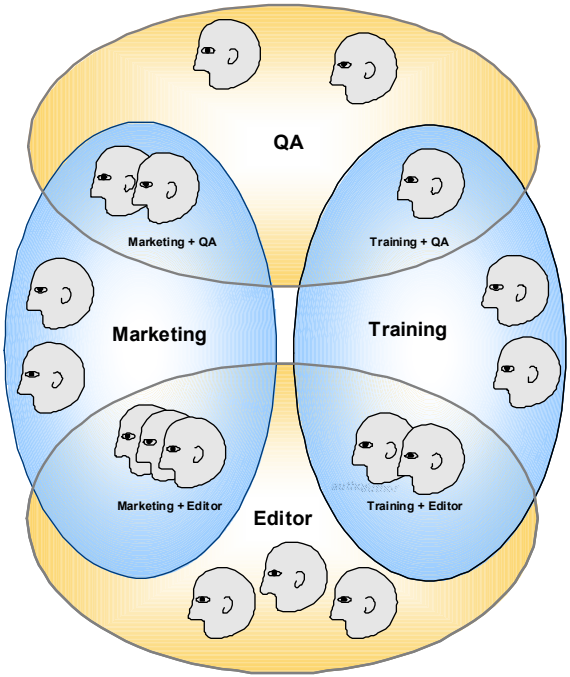


Fig. 9 – Assignment of users to groups and roles

By using group-role settings, you can set very specific access rights for objects, i.e. both subject-specific (by using the group) and task-specific (by using the role). For example, the access rights for the topic area “Marketing” could be set in such a way that only editors who are assigned to the group “Marketing” are given the rights “Create”, “Change object”, “Change metadata”, etc. Similarly, the right to release objects could be assigned to the group-role “Marketing + QA” and the right to change access rights to the group-role “Marketing + ProjectManager”:

Table 5 – Example of group-roles and rights

Assignment of groups and roles	Rights for the topic area “Marketing”
Group “Marketing” + Role “ProjectManager”	Read + Change rights + Change metadata
Group “Marketing” + Role “Editor”	Read + Create + Change object + Change metadata + Delete
Group “Marketing” + Role “QA”	Read + Release

Default Access Right Settings

In the Admin client, the WCM administrator can define default access rights for each user, group, and role. When a principal is added to the list of those authorized to access an object in the Content client, these default settings are used as the initial settings for the access rights. You can change these presettings at any time in the Access rights dialog box.

Note: We recommend that you make as few changes to the access rights as possible at individual object level. That way, you prevent the rights actually granted from becoming too confusing.

Interpreting Access Rights

The basic rule is that users may only access an object if they have the necessary access right. Thus, it is not necessary to forbid access explicitly. As long as access is not explicitly permitted or not inherited from a parent topic, it is forbidden by default.

If you want to allow only a particular user group to work in a topic area, it is not necessary to deny access to the objects in this area to all other groups. Simply assign the relevant access right to the respective group. For all others, the following rule applies: what is not explicitly permitted is forbidden.

Inheritance of Access Rights

The access settings for a topic are automatically transmitted to the pages, topics, and other objects added to this topic. This transmission of access right settings to child objects is known as *inheritance*. Thus, the access settings of a topic are inherited by all objects below that topic.

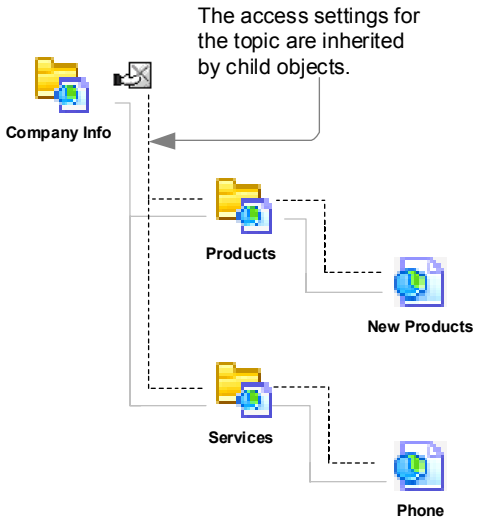


Fig. 10 – Inheritance of access rights

The inheritance of access right settings by subordinate objects enables you to set the access rights for a website without great efforts. You can grant different user groups access to the topics and objects in the website in accordance with their individual tasks. Every new object inherits the access rights of the topic to which it belongs. Thus, if you have granted certain groups, roles, or users access to a topic, they will have the same access rights for any new pages or topics added to the parent topic.

The access rights inherited from the topic can be changed at any time. Such changes to access rights can, however, only be made by users who have the access right “Change rights”.

Any change to the access rights of an object interrupts the chain of inheritance from the parent topic. If the changed object is a topic, a new chain of inheritance starts at this point. The objects added to this topic inherit the new, changed access settings.

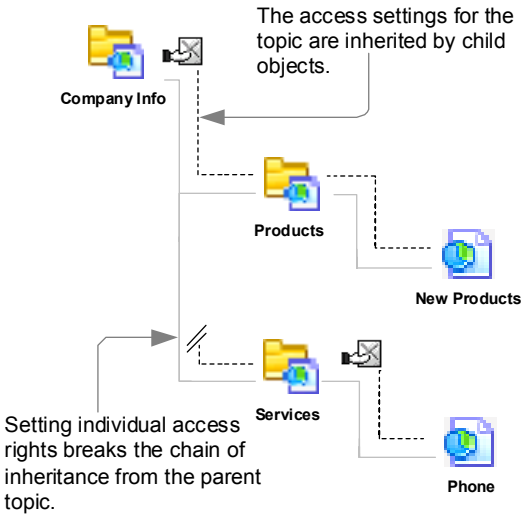

The access settings for the topic are inherited by child objects.
Company Info
Products
New Products
Services
Phone
Setting individual access rights breaks the chain of inheritance from the parent topic.

Fig. 11 – Interrupted chain of inheritance as a result of changed access rights

If you subsequently change the access rights for a topic, this alteration affects not only the topic itself, but also all objects in the topic's chain of inheritance. The change has no effect on the objects in a different chain of inheritance.

In order to discard the changed access rights and reintegrate the object in the chain of inheritance of the parent topic, you can apply the function "Use access rights of parent topic".

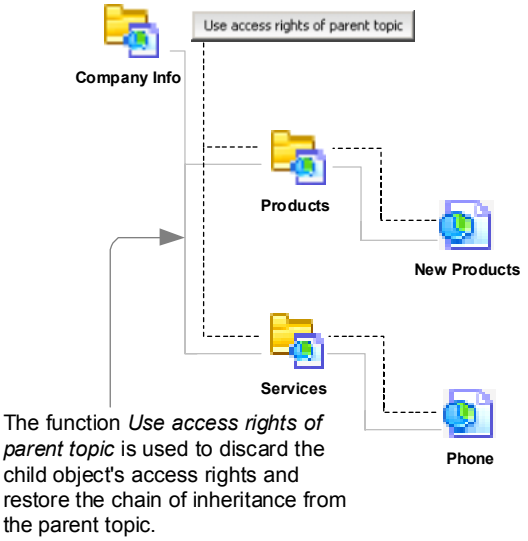

Use access rights of parent topic
Company Info
Products
New Products
Services
Phone
The function *Use access rights of parent topic* is used to discard the child object's access rights and restore the chain of inheritance from the parent topic.

Fig. 12 – Restoring the inherited access rights

Priority Rules

A user can be authorized to access an object and at the same time belong to a group or role also having access to the object. The actual access rights of the user result from the combination of the group-specific or role-specific access rights with the user-specific access rights. Since the access rights for an object may be defined differently for groups, roles, and individual users, it is possible that this combination may result in conflicting access rights. Such conflicts are resolved by means of the following priority rules:

1. Explicit prohibition takes priority over explicit permission.
2. An individual explicit permission overrides a general or group/role-specific prohibition.
3. An individual explicit prohibition overrides an explicit group/role-specific permission.
4. If an access right is neither explicitly permitted nor forbidden, an implicit prohibition applies, i.e. access is not possible.

The following simple example illustrates these priority rules.

Example

The user “jstein” is a trainee in the “Marketing” department. He is given the task of editing various Internet pages, including the list of new products and the overview of all products.

In the Admin client, “jstein” is assigned to the group “Marketing” and the role “Trainee”. This gives him access rights to the objects below the topic “Products”. Because of his group and role membership, the access rights are set as follows:

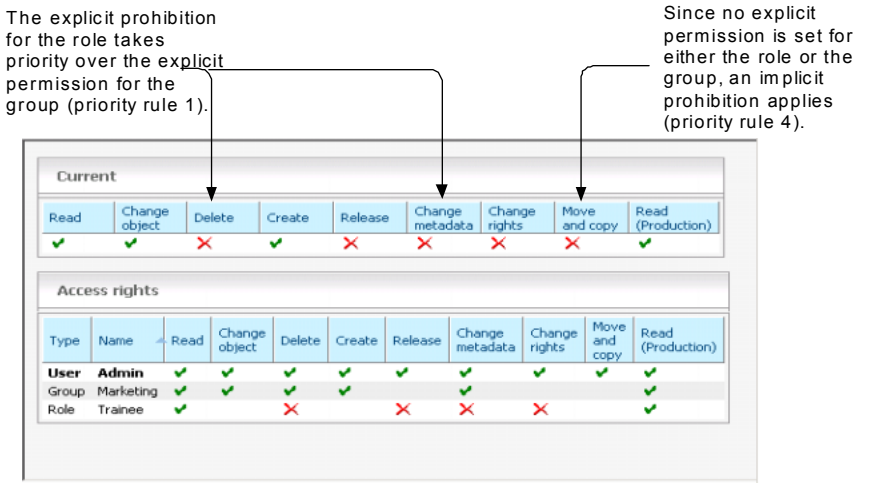


Fig. 13 – Interpreting group- and role-specific access rights

“jstein” has the task to regularly update the various pages under “Products”. At the same time, “jstein” must also make sure that an appropriate expiration date is set for each page. If necessary, he must set the expiration date in the metadata.

The access rights for the role “Trainee” are set in such a way that the right “Change metadata” is denied. This explicit prohibition to change the metadata also takes priority over the explicit permission for the group “Marketing”. Thus, the result of this setting is that “jstein” cannot change the metadata of the pages.

There are several ways of giving the user “jstein” the access right required for changing metadata:

- Remove the user “jstein” from the role “Trainee”. In that case, his access rights would be determined entirely by his membership in the group “Marketing”. Because of this group membership (without the role-specific restriction), he would, however, have not only the right to change metadata, but also the right to delete objects.
- Change the access settings for the role “Trainee” so that changing metadata is not explicitly forbidden. This, however, would apply to all trainees, not just to “jstein”.
- Give the user “jstein” explicit permission to change metadata. User-specific permission of this kind overrules the prohibition resulting from assignment to the role.

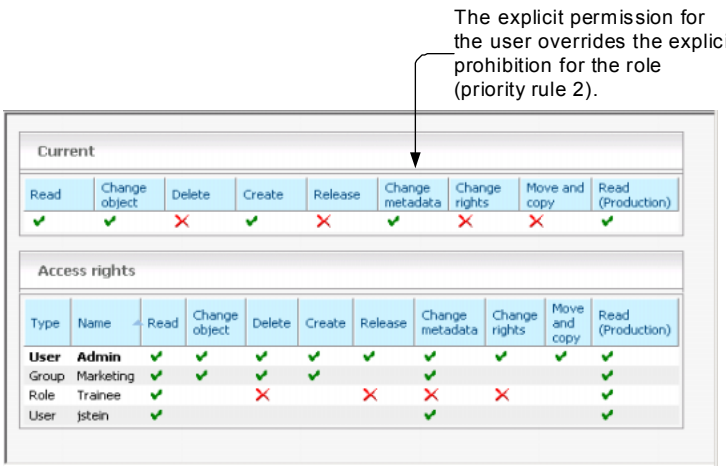


Fig. 14 – Interpreting group-, role-, and user-specific access rights

Note: This example is merely intended to illustrate the effects of the various priority rules; it is not designed to be a model for optimum access right settings.

Functional Areas

Functional areas control which functions are available to the users in the Content client. They fulfill two different tasks:

- Functional areas determine which types of objects a user may add, check out, and check in. The functional areas such as “Basic” are by default linked with object types. Only a user who has the corresponding functional area can add objects of this object type.
- Functional areas determine which views and dialog boxes are available to the user in the Content client. This way, you can determine exactly which actions the user is allowed to perform. For example, the Access rights dialog box is only displayed if the user has the functional area “Access rights dialog”. If the user is not assigned this functional area via group or role membership (or a direct assignment), the user cannot view and edit the access rights of the WCM objects.

> **Note:** The layout of your Content client strongly depends on the functional areas you have been assigned and can thus be different from the description in this manual.

In the Admin client, functional areas can be assigned to groups, roles, or individual users. Also, the WCM administrator can add functional areas and change the default assignment of object types. The following table gives an overview of the default functional areas and their assignment to object types and to functions and views in the Content client. For some Content client messages, the internal names of the functional areas are used. For this reason, they are also listed in the table.

Table 6 – Functional areas and object types

Functional area (Internal Name)	Default assignment to object type	View or function in the Content client
Basic (VIP)	Compound object, Excel document, GIF image, HTML page, JPEG image, Other, PDF document, PNG image, PowerPoint file, Topic, Word document, XML document	Add, check out, and check in objects based on these object types
Advanced (ADVANCED)	Frame, Frame topic, HTML template, XML template	
Dynamic (DYNAMIC)	Assembled object, ASP page, ASP template, ASP topic, CGI script, Form template, JavaScript, JSP page, JSP template, JSP topic, PHP page, PHP template, PHP topic, XSLT document, XSLT template	
Form (FORM)	Form instance	
Workflow (WORKFLOW)	Workflow	Assign workflows to objects, remove workflow assignments

Functional area (Internal Name)	Default assignment to object type	View or function in the Content client
Intelligent templates (ITF)	none	Not used by default, required for compatibility with VIP 5e
Direct release (DIRECT_RELEASE)		Work with the “Direct release” option.
References dialog (REFERENCES)		View the References dialog box
Access rights dialog (ACCESS_RIGHTS)		View the Access rights dialog box
Log dialog (LOG)		View the Log dialog box
Livelink (LIVELINK)		View the Livelink metadata dialog box
Filter standard (FILTER_STANDARD)		Use the standard filters

Functional area (Internal Name)	Default assignment to object type	View or function in the Content client
Filter edit (FILTER_EDIT)	none	Create and edit filters in the filter editor **Note:** Additionally, the functional area “Filter standard” is required.
View Subordinate objects (OBJECTLIST)		View the subordinate objects of a WCM object
View Object list (LISTVIEW)		“Object list” view
View My objects (FILTER_TODO)		“My objects” and “My work list” views
View Template structure (TEMPLATE_STRUCTURE)		“Template structure” view
Import (IMPORT)		Use import functions
Search (COMI_SEARCH)		Use search functions

Object Types

This section introduces the object types used in Livelink WCM Server. Each WCM object is assigned to a specific object type. The object type is defined when the object is added. There are only a few cases in which it may subsequently be changed. You can manage object types via the *Tools* menu of the Content client or in the Admin client.

An object's type depends on its content and on its use. For example, there is no difference between the content of an object of type "Topic" and an object of type "HTML page". The topic, however, has an additional organizational function in the object hierarchy because it can accommodate other (child) objects.

The following table provides an overview of the main features of the standard object types.

> **Note:** The choice of object types you are offered when adding an object may vary from the following table. Object types are linked with functional areas, i.e. you can only see the object types for which you have the respective functional areas. Also, your WCM administrator may delete and/or add object types.

Table 7 – Object types with description

Object type	Icon	Description
Assembled object		This object type is used to integrate objects that do not have a content component themselves. The complete object information is saved in the metadata. This way, the content of other objects can be integrated in the objects. For information on using this object type, refer to chapter 10 “Working with Assembled Objects”.
ASP page		This object type is used to integrate objects whose content includes ASP scripts.
ASP template		In addition to page layout description, this object type makes it possible to reuse routines that have been programmed once for the individual creation of web content without difficulty in other objects.
ASP topic		This object type represents a combination of an ASP object and a directory. As with other topic types, an ASP topic can contain subordinate objects.
CGI script		This object type is used to integrate executable files.
Compound object		This object type is used for downward compatibility and represents complex document structures that are to be managed as a single document.
CSS		This object type is used to integrate Cascading Style Sheets (CSS).
Excel document		This object type is used to integrate Microsoft Excel documents (e.g. spreadsheets).

Object type	Icon	Description
Form instance		This object type is used for creating forms. It is represented by an XML file.
Form template		This object type defines the type “Form”, i.e. it is used to identify and categorize the different form instances.
Frame		Frames can be used to divide the browser window into separate, freely definable segments. Each of these segments can have content of its own. The individual segments (i.e. frames) may have static or dynamic content.
Frame topic		This object type represents a combination of a frame object and a directory. As with other topic types, a frame topic can contain subordinate objects.
GIF image		This object type is used to integrate graphics in the GIF file format.
HTML page		This object type is used to integrate HTML pages which may include text, hyperlinks, and graphics.
HTML template		An HTML template is an HTML page that describes the layout of the pages. Special placeholders are used in templates to integrate the content items and (optionally) metadata of those objects that use a template. The content of the WCM objects is automatically merged with the template used.
JavaScript		This object type is used to integrate JavaScript files.
JPEG image		This object type is used to integrate graphics in the JPEG file format.

Object type	Icon	Description
JSP page		This object type may contain at least one script that can be executed on a web server. Such scripts can be used to dynamically compose or compute web content items and then display them.
JSP template		In addition to page layout description, this object type makes it possible to reuse routines that have been programmed once for the individual creation of web content without difficulty in other objects.
JSP topic		This object type represents a combination of a JSP object and a directory. As with other topic types, a JSP topic can contain child objects.
Livelink folder relator		This object type is used to integrate folders from a Livelink system.
Livelink relator		This object type is used to integrate documents from a Livelink system.
Other		This object type is used to integrate documents that do not belong to any of the other object types. Examples include graphics in formats other than GIF, JPEG, and PNG.
PDF document		This object type is used to integrate PDF files.
PHP page		This object type is used to integrate PHP pages.
PHP template		This object type represents a PHP document that is used as a template.

Object type	Icon	Description
PHP topic		This object type represents a combination of a PHP object and a directory. As with other topic types, a PHP topic can contain child objects.
PNG image		This object type is used to integrate graphics in the PNG file format.
PowerPoint file		This object type is used to integrate Microsoft PowerPoint documents.
Script		This object type is used to integrate HTML-like documents without head or body section.
Topic		This object type represents a combination of an HTML page and a directory. A topic can be edited like any other HTML page. You can insert hyperlinks and graphics on the page. At the same time, topics can contain child objects and can thus be used to structure the website.
Word document		This object type is used to integrate Microsoft Word documents.
Workflow		This object type is used to integrate workflow definitions in XPDL format created with the Content Workflow Modeler. For information on using this object type, refer to chapter 8 “Content Workflow”.
XML document		This object type is used to integrate documents in the format “Extensible Markup Language” (XML). For information on using this object type, refer to appendix A “Livelink WCM Server and XML”.

Object type	Icon	Description
XML schema		This object type is used to integrate XML schema objects.
XML template		This object type represents an XML document that is used as a template. By means of an XML template, it is, for example, possible to provide a uniform framework at the data level for data that is stored in XML format. For information on using this object type, refer to appendix A “Livelink WCM Server and XML”.
XSLT document		This object type is used to integrate documents with information on XSL transformations. These transformations describe how to convert an XML document into, for example, an XHTML document. The conversion is not restricted to XHTML documents. Free text-based formats are also possible. For information on using this object type, refer to appendix A “Livelink WCM Server and XML”.
XSLT template		This object type represents an XSLT document that is used as a template. An XSLT template contains transformation rules. These rules are applied to the XML documents that have been assigned the XSLT template. The transformation is performed on the server side. For information on using this object type, refer to appendix A “Livelink WCM Server and XML”.

CHAPTER 3

Working with the Content Client

The Content client is used to create, edit, check, and publish the content of a website. This chapter shows you the first steps in working with the Content client and introduces you to the user interface and the log. A detailed description of the functions of Livelink WCM Server can be found in the following chapters.

In order to work with the Content client and make editorial changes to a website, you do not need to install anything on your workstation. All you need is:

- a **browser**. Configure the browser in such a way that
  - JavaScript is activated
  - non-persistent cookies are accepted. (To be able to make use of profiles, your browser must also accept persistent cookies.)
  - encoding is set to Auto-Select or to the encoding of the Content server which you access with the Content client (e.g. Unicode/ UTF-8)
  - it allows additional browser windows – so called popup windows – to open

> **Note:** If you work with the operating system Windows XP, please switch off ClearType font smoothing. Right-click on the desktop and choose *Properties*. Select the *Appearance* tab and click the *Effects...* button. Mark the check box next to *Use the following method to smooth edges of screen fonts* and click “Standard”.

- the **right to access** the WCM system. You can obtain your user ID and a password from your WCM administrator.
- **editor programs**: You can use your familiar programs to edit the website content, i.e. any HTML editor for HTML pages, your preferred graphics program for graphics, etc. For editing HTML objects, the Content client offers an integrated HTML editor.

Starting the Content Client

This section describes the first steps in working with the Content client. The following procedures will be explained:

- log in to the WCM system, see the following section
- select the website you want to edit, see section “Selecting a Website” on page 76
- log out of the WCM system, see section “Logging Out of the WCM System” on page 77
- change your password, see section “Changing the Password” on page 78
- log in as substitute for another user, see section “Taking over as Substitute” on page 79
- display your login information, see section “Information on the Logged-in User” on page 80

Logging in to the WCM System

Before you log in to the WCM system for the first time, you will receive the following information from your WCM administrator:

- the URL of the Content client
- you user ID and the corresponding password

To log in to the WCM system:

1. Launch your browser.
2. Enter the URL of the Content client in the address field of your browser.

   The *Log in* dialog box opens.

Fig. 15 – Logging in to the Content client

3. Enter your user ID and the corresponding password.

   Note: User ID and password are case sensitive. For the WCM system, “user” and “User” are two different users.

4. Click the *Log in* button.

   When you log in to the WCM system for the first time, you are asked to change your password.

Selecting a Website

After you have successfully logged in to the WCM system, the *Select website* dialog box is displayed. This dialog box gives you an overview of all websites you can access.

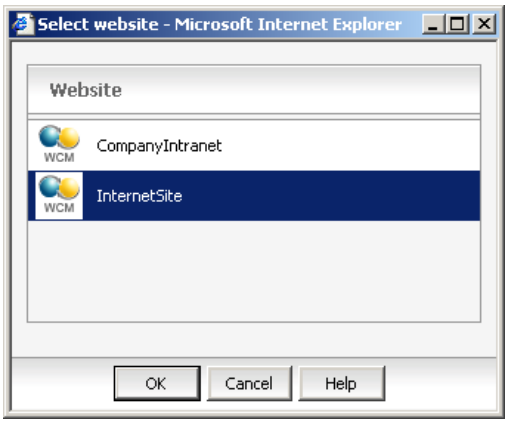


Fig. 16 – Selecting a website

To determine the website that you want to work with:

1. In the selection list, mark the desired website.
2. Confirm by clicking the *OK* button.

   The name of the selected website is displayed in the header.

After login, the view you selected last is displayed by default. When you log in for the first time, the Edit view is displayed. You can select any other available view at any time via *Tools* → *View*.

- The Edit view is available as soon as a website has been created in the Admin client and read rights have been set.

- The QA view is available if the topmost topic of the website – the so-called root object – has been submitted at least once.
- The Production view is available if the root object has been released.

Of course, you can change to another website while working with the Content client. To do so, choose *Select website* on the *Tools* menu.

Logging Out of the WCM System

> **Note:** There are certain time-consuming actions in Livelink WCM Server that are performed asynchronously, i.e. the actions are automatically completed in the background. Time-consuming actions include importing objects and deleting topics. In these cases, just close your browser to stop working with the Content client. Logging out will cancel the actions.

To log out of the WCM system:

1. Choose *Object* → *Exit*.
2. Confirm the security prompt by clicking the *Yes* button.

Licenses

As soon as you have logged in successfully to the WCM system and perform a staging action in the Content client, you use up a license.

To stop working with the Content client, you should always choose *Object* → *Exit* before closing the browser. Otherwise, you will remain internally logged in to the WCM system for a certain period of time (this period is set in the Admin client). The license you are using will not be released until this time has elapsed.

Changing the Password

For security reasons, we recommend that you change the password for accessing the WCM system at regular intervals.

To change your password:

1. Choose *Tools* → *Change password*.

   This opens the *Change password* dialog box.

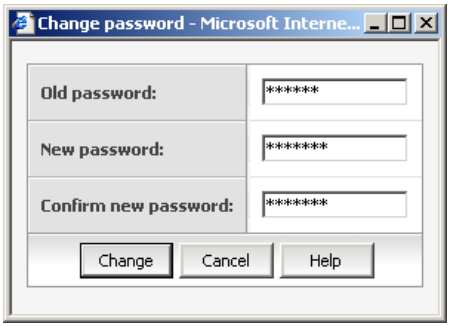


Fig. 17 – Change password

2. Enter your old password once and the new password twice.

   Note: Please note that passwords are case sensitive. For the WCM system, “password” and “Password” are two different passwords.

3. Click the *Change* button.

Taking over as Substitute

The substitute of a user may perform all functions for which the user is authorized. To be able to edit objects as a substitute for a user, you have to be assigned as a substitute to this user in the Admin client.

To take over as substitute:

1. Choose *Tools* → *Take over as substitute*.

   This opens the *Take over as substitute* dialog box.

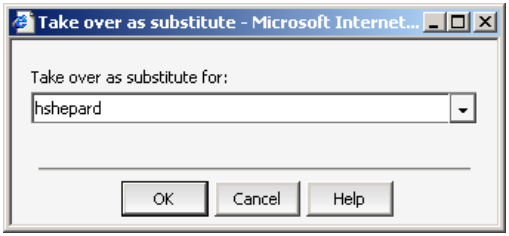


Fig. 18 – Taking over as a substitute

2. In the drop-down list, click the ID of the user for whom you want to act as substitute.
3. Confirm by clicking the *OK* button.

   The status bar displays the name and ID of the user for whom you took over as substitute.

Notes:

As a substitute, you cannot take over as another substitute. Example: The user “jstein” is the substitute of “mmoreno”, who is entered as substitute for “cerdmann”. “jstein” takes over as the substitute for “mmoreno”. However, as “mmoreno”, jstein **cannot** take over as the substitute for “cerdmann”.

To work with your own user data after having acted as substitute, exit the Content client and log in again with your user data.

Information on the Logged-in User

The Content client provides a dialog box showing all information on the logged-in user. To open this dialog box, choose *Tools* → *Login info*.

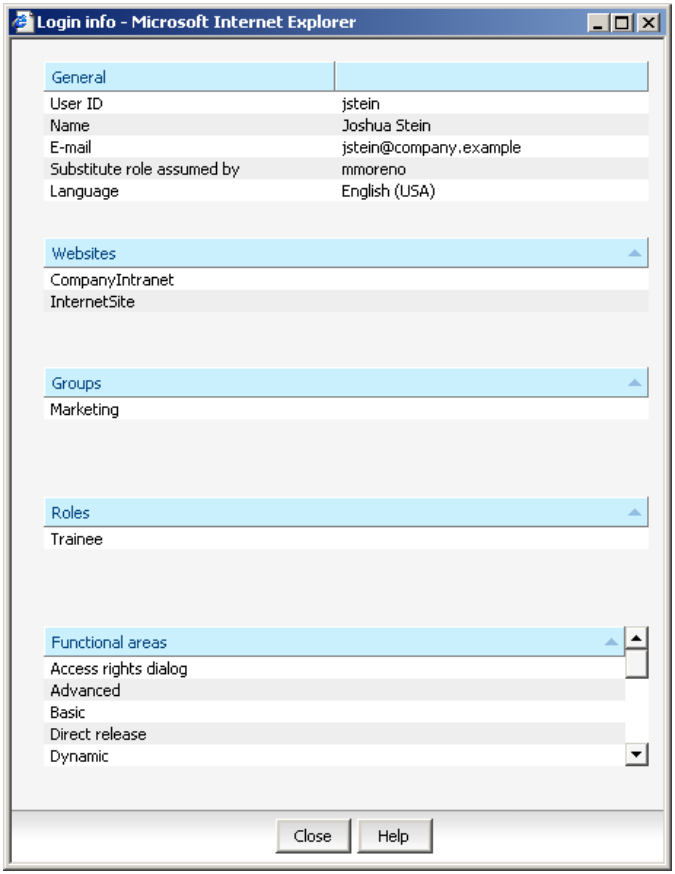


Fig. 19 – The Login info dialog box

The dialog box contains all information on the logged-in user that is available in the WCM system. In addition to the general user information (name, user ID, e-mail address, and language setting), you can see at a glance which websites, groups, roles, and functional areas you are assigned to. The dialog box also shows whether you are logged in as the substitute of another user. The displayed data is managed via the Admin client and can only be viewed here.

The User Interface of the Content Client

After you have logged in successfully and selected a website, the user interface of the Content client builds up. The window is divided into different areas.

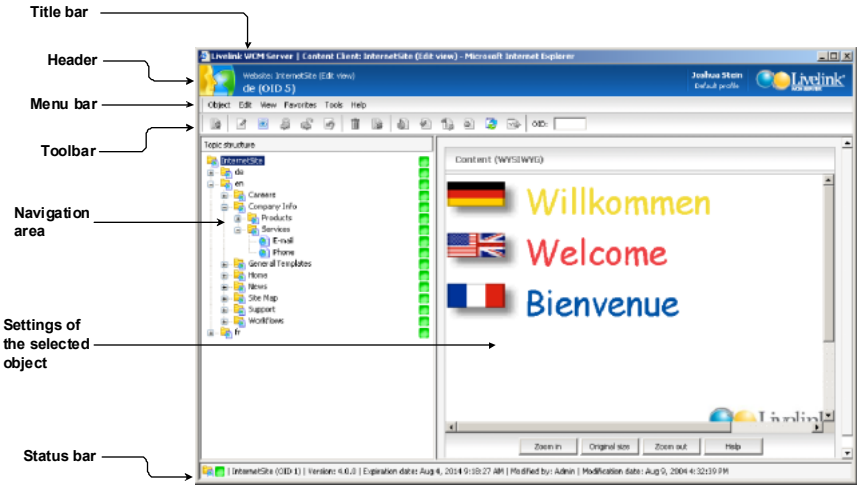


Fig. 20 – The user interface of the Content client

The following sections provide a short introduction to the user interface of the Content client. It is explained how to

- use the Content client to navigate through the website, see the following section
- call functions in the Content client, see section “Choosing Functions” starting on page 85

Navigating in the Content Client

You can use the navigation area to quickly navigate within the website. Via the navigation area, you can find pages or graphics and select them for editing. The selected object is highlighted in color.

Another navigation tool is the *OID* field in the toolbar. If you want to edit an object and know its OID, enter the corresponding number here. This takes you straight to the desired object.

The Content client provides different views of the website structure. In the views, the objects of your website are organized according to different criteria. They are displayed in an object tree or an object list. Via the *View* menu, you can toggle between different views.

> **Note:** The available views depend on your functional areas.

The “Topic structure” view

Choose *View* → *Topic structure* to open the topic structure. The topic structure is displayed as an object tree. The topmost node of the object tree is the name of the website. Starting from this node, the object tree branches to topics and subtopics, like the directories and subdirectories in a file system.

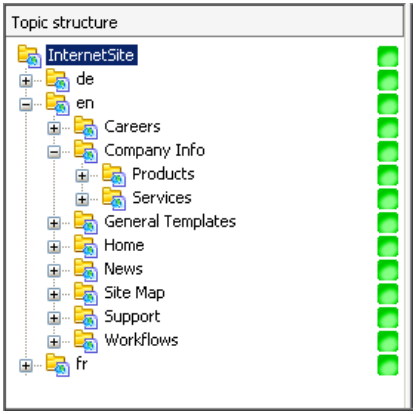


Fig. 21 – The “Topic structure” view

If a topic contains subordinate objects, the topic node is displayed as [+] or [-]. Click [+] or [-] to open or close the relevant topic branch and view the subordinate objects.

In the Configuration dialog box, you can determine how the nodes look and specify the maximum number of objects to be displayed below a topic.

The “Template structure” view

Choose *View* → *Template structure* to open the template structure. The template structure is displayed as an object tree. This view displays the various templates used in the website in alphabetical order.

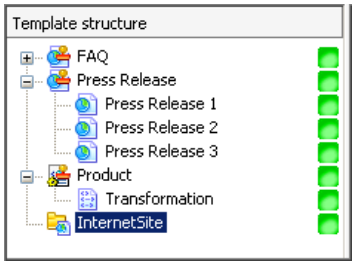


Fig. 22 – The “Template structure” view

If a template has been assigned to other objects, the template node is displayed as [+] or [-]. Click [+] or [-] to show the objects using this template. Thus, if you want to change a template, you can see at a glance which objects would be affected by this change.

In the Configuration dialog box, you can determine how the nodes look and specify the maximum number of objects to be displayed below a template.

The "Object list" view

Choose *View → Object list to open the object list.* The object list displays all objects directly subordinate to a topic. For each object, the object type, OID, title, version, and status are displayed. By clicking on the header row of a column, you can sort the object list according to the criterion in this column. For example, clicking on the *Title* column sorts the list by title in ascending alphabetical order. To reverse the sort order, click the header row again. This way you can clearly arrange long lists.

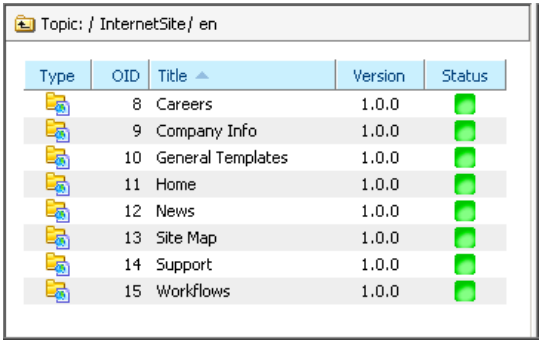


Fig. 23 – The "Object list" view

Above the object list, the topic whose object list you are viewing is displayed. To navigate up in the object hierarchy, click the icon or the respective entry in the path. If you want to navigate down in the object

hierarchy, double-click an entry in the list. This opens the object list of the respective object.

In the Configuration dialog box, you can determine the maximum number of entries to be displayed in the object list.

The Favorites list

For quick access to objects, you can use the personal favorites list. The *Favorites* menu takes you straight to the objects you frequently edit. Favorites are only displayed for the currently selected website.

To add an object to the list of favorites, mark the object in the navigation area and choose *Favorites* → *Add to Favorites*. Choose *Favorites* → *Remove from Favorites* to delete the current object from the list of favorites.

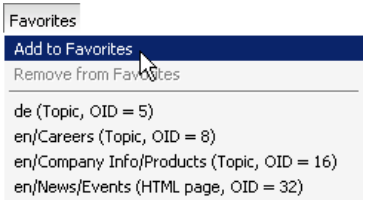

Favorites
Add to Favorites
Remove from Favorites
de (Topic, OID = 5)
en/Careers (Topic, OID = 8)
en/Company Info/Products (Topic, OID = 16)
en/News/Events (HTML page, OID = 32)

Fig. 24 – The Favorites list

Choosing Functions

In the Content client, you can choose functions in the following ways:

- from the menu bar
- from the context menu
- from the toolbar
- from the “Subordinate objects” view

The menu items and icons are dynamically modified to suit the current context. Functions that you cannot use on the object currently selected in the navigation area are deactivated (dimmed).

Editing Objects in the Content Client

This section contains a short introduction to working with the Content client. It is explained

- how to select an object, see the following section
- which possibilities exist to display an object, see section “Displaying Objects” on page 88
- how to send e-mails when the status of an object changes, see section “Sending E-Mails” on page 92
- how to find objects within the website, see section “Filter and Search Functions” on page 96
- how to track the progress of time-consuming actions, see section “The Action List” on page 101

Selecting an Object

To edit an object, mark it in the navigation area. If the Content client is configured accordingly, the object information of the marked object, the so-called *current object*, is displayed in the left area of the status bar.

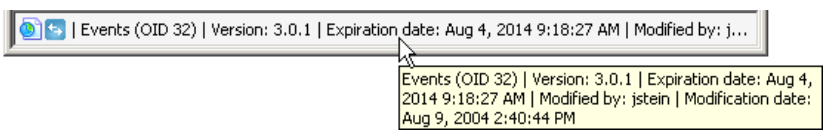

| Events (OID 32) | Version: 3.0.1 | Expiration date: Aug 4, 2014 9:18:27 AM | Modified by: j...
Events (OID 32) | Version: 3.0.1 | Expiration date: Aug 4, 2014 9:18:27 AM | Modified by: jstein | Modification date: Aug 9, 2004 2:40:44 PM

Fig. 25 – Object information in the status bar

The following data from the object information is shown in the status bar, providing you with a quick overview of the most important properties of the current object:

- object type as icon (here: HTML page)
- object status as icon (here: changed)
- object title (here: Events)
- OID (here: 32)
- version number (here: 3.0.1)

  The three-digit version number consists of the numbers for major version, minor version, and micro version.

  - The major version (here: 3) is incremented by 1 whenever the object is released.
  - The minor version (here: 0) always has the value 1 if the object has the status “submitted”. Otherwise the value is 0.
  - The micro version (here: 1) changes with every action in the Edit view (e.g. “Check out”, “Check in”)

  Thus, an object with the version number 3.0.1 has passed through all staging steps three times and has been released in the Production view. The object was edited, but the changes have neither been submitted to QA nor published in the Production view.
- date and time until which the object remains valid (here: 04.08.2014 09:18:27). Expired objects are marked by red letters in the status bar.
- the ID of the last user who changed the object (here: jstein)
- date and time of the last change to the object (here: 09.08.2004 14:40:44)

Selecting several objects

To select several objects for editing in the “Object list” view or in a filter hitlist:

- Mark the objects to be edited with the left mouse button while, **at the same time**, pressing the CTRL key.
- A group of objects located directly below each other is selected by marking the highest and the lowest object while, **at the same time**, pressing the SHIFT key.

Displaying Objects

In the Content client, there are different display options for the object you selected in the navigation area:

- displaying the object content, see the following section
- displaying the object preview, see section “Displaying the Object Preview” on page 89
- displaying the object preview on the basis of a dynamic deployment system, see section “Object Preview on the Basis of a Dynamic Deployment System” on page 90
- displaying the object content in the editor, see section “Displaying the Object Content in the Editor” on page 92
- displaying the assigned workflow, see section “Displaying Information on the Workflow” on page 92

Displaying the Object Content

The *View* menu provides two modes for displaying the content of an individual object.

- *Content (WYSIWYG):* In this mode, you can view the content in the right window pane as it is displayed in the browser. You will only see the actual content of the object, i.e. the information from the template is not adopted and the WCM tags are not replaced. Zoom functions are provided for a convenient use of this view.

  Notes:

  When viewing objects that have dynamic content or use JavaScript, errors may occur in this mode that do not occur in the generated pages.

  The Mozilla browser does not support the zoom functions.

- *Content (source):* In this mode, you can view the object content in the right window pane as plain text.

In the Configuration dialog box, you can determine which object types are to be displayed in the modes.

Displaying the Object Preview

When you log in to the WCM system, a browser window for the object preview always opens automatically. The object preview displays the page that the current deployment system generates for the object selected in the navigation area. If you have, for example, choosen *Edit view* under *Tools* → *View*, you see the view of the object generated by an Edit deployment system

Choose *View* → *{name of the deployment system}* to display the view of a different deployment system in the object preview (the current deployment system is checked in the menu). This way you can easily check how the object that you edited currently looks in the QA or Production view.

Note: If you select different objects in rapid succession, the object preview may no longer by synchronized with the Content client. In this case, select the current object again.

Object Preview on the Basis of a Dynamic Deployment System

Deployment systems generate pages for the WCM objects stored in the database. These pages can be displayed in a browser. *Dynamic deployment systems* do not generate the pages statically when the WCM object changes, but dynamically when the page is requested by the HTTP server. If a changed object is accessed, e.g. because a user wants to edit it with the Content client, the page is processed by the deployment system. Thus, the current version of the object is always displayed. Dynamic page generation is especially advantageous if many changes are made to the WCM objects, e.g. for the Edit view.

Dynamic deployment systems can generate pages for different versions of the WCM objects:

- for the different views, i.e. Edit, QA, and Production
- for different template versions (released template or current template in the Edit view)
- for the object version at a certain point in time
- for the object state with the changes of certain users

  This option can, for example, be used by template developers to view how the generated pages will look with templates that they are currently editing. Only the changes of the current user are taken into account. Additionally, all changes that other users made before the object was edited by the selected user are displayed.

In this way, you get a personalized view of the website.

In the Configuration dialog box, you can determine which rendition of the WCM objects you want to display.

Example: Releasing a master template

The website has the following template cascade: the template “Department” is based on the template “DepartmentStore”, which is based on the template “Company”. The highest-level template in the cascade (the template “Company”) has been changed. It is your task to check these changes and their effect on the dependent templates and objects, and release them afterwards.

This presupposes that a dynamic deployment system has been configured for the website in the Admin client. (For further information, refer to the online help of the Admin client.)

Proceed as follows:

1. Choose *View* → *{name of the dynamic deployment system}*.
2. Choose *Tools* → *Configuration* to open the Configuration dialog box.
3. In the Configuration dialog box, make the following settings for the dynamic deployment:
   - *Deployment view*: Production view
   - *Template view*: Current template

   The published version of the objects on the basis of the changed template is displayed in the object preview.
4. Browse through the website and check the effects of the changed template “Company”.
5. After you have checked the changes, you can release or reject the template.

Displaying the Object Content in the Editor

Choose *Object* → *Display content* to open the object in the editor you have specified for objects of this type in the Configuration dialog box. This action does not change the status of the object. This means that the object is not barred for write access by other users. The action is not written to the log.

Displaying Information on the Workflow

Choose *View* → *Workflow* to display the following information on the workflow that is assigned to the current object:

- name and description of the workflow, as well as name and OID of the object containing the workflow definition
- name, assigned principal, and description of the current activity
- list of all transitions of the current activity

Additionally, a graphical representation of the workflow definition is displayed in the lower window pane. The current activity is highlighted by a green frame.

Sending E-Mails

After you have edited an object, you can inform the person responsible for the next work step by e-mail. The e-mail addresses of the responsible persons are stored in the metadata of the object.

> **Note:** If a workflow is assigned to the object, the system automatically sends an e-mail to the address of the user who is assigned to the next workflow step. In the case of a staging transition, no separate e-mail is sent to the address specified in the metadata.

Sending an e-mail is possible after the following actions:

Table 8 – Staging steps and e-mail notification

Staging step	Notified user
Submit	E-mail address from the metadata item “E-mail QA”
Release	E-mail address from the metadata item “E-mail Release”
Reject	E-mail address from the metadata item “E-mail Edit”
Delete released object	E-mail address from the metadata item “E-mail QA”
Destroy release object	E-mail address from the metadata item “E-mail Release”

If you have performed one of these actions, the following E-mail dialog box opens. For detailed information on the individual fields, refer to the Content client's online help.

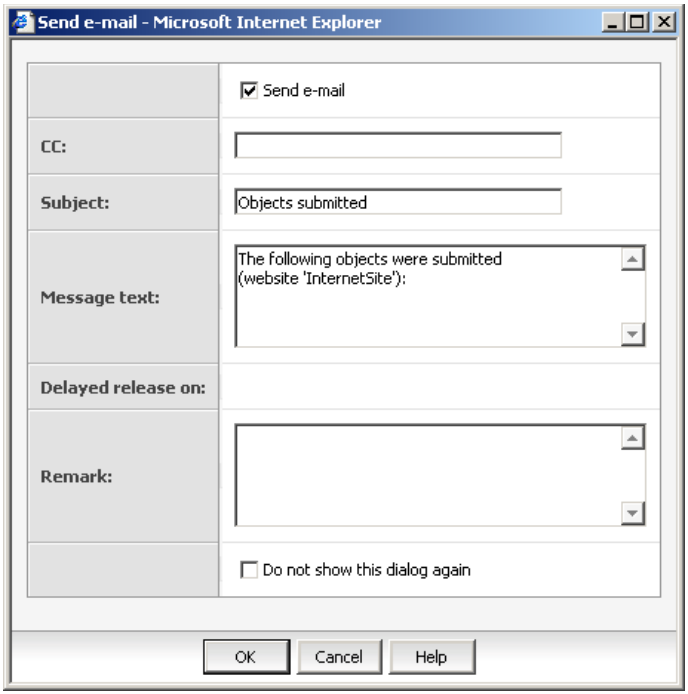


Fig. 26 – Sending an e-mail

Activating links to the Content client for e-mails

The WCM system can be configured in such a way that the e-mails sent contain links to the edited objects. If you select one of these links, the Content client starts. After the login, the respective WCM object is selected in the navigation area.

To activate this function, you must modify the file **{WCM installation directory}\config\mail-conf.xml** accordingly. For each Content server, a separate `<mailurls>` section must be added with the following entries.

> **Notes:**
>
> Only a WCM administrator should make changes to the **mail-conf.xml** file.
>
> For the changes to take effect, all servers must be restarted.

Table 9 – Activating links for e-mails in the mail-conf.xml file

Line in the XML file **Explanation**
`<type name="DynamicWithLinks"/>` Activates the insertion of links in the e-mails
`<mailurl name="Master_Content">` The name of the Content server sending the e-mails
`<cms_url>http://wcmserver.company.example/cmsclient/index.jsp </cms_url>` The URL to the Content client
`<cms_display_name>Content-Server Chicago</cms_display_name>` Freely definable display name for the server in the e-mail. In a distributed WCM system, you may, for example, specify the location of the server.

Filter and Search Functions

The Content client provides extensive filter and search functions, which help you find objects quickly.

> **Note:** The filter and search functions displayed in your Content client may be different from the following description. They depend on your functional areas.

The “Object filter” view

Choose *View* → *Object filter* to open the object filter. This filter helps you find and list objects on the basis of specific criteria (including user-defined criteria). On the filtered objects, you can perform actions such as “Submit” or “Release”. This way you can edit several objects simultaneously.

The filter result is displayed in an object list. For each object, the object type, OID, title, path to the root object, and status are displayed. By clicking on the header row of a column, you can sort the object list according to the criterion in the column. Another click reverses the sort order. You can mark one or more objects in the list in order to edit them.

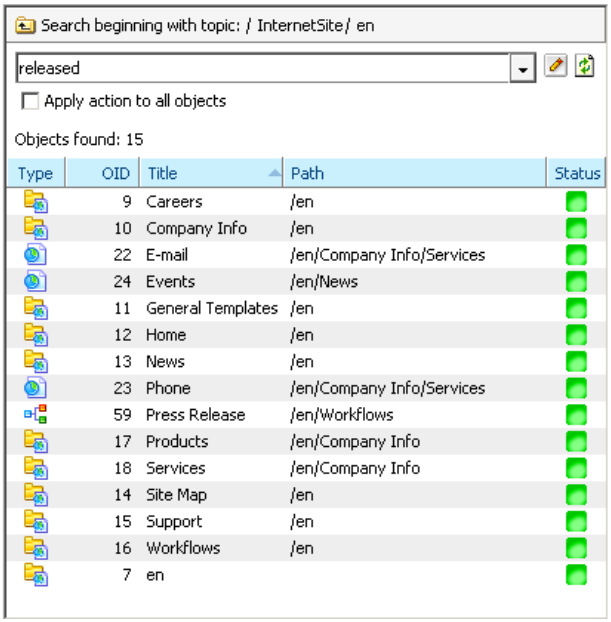


Fig. 27 – The “Object filter” view

For detailed information on the functions and use of the object filter, refer to chapter 12 “Finding Objects and Editing Them Together”.

The “My objects” view

Choose *View* → *My objects* to open the “My objects” view. This view is personalized as it considers your access rights. Only those objects that you have edited or that you can edit are displayed. The view offers a list from which you can select the criteria for filtering the website objects. For example, if you click *I have changed* in the drop-down list, the object list only shows those objects that you changed last.

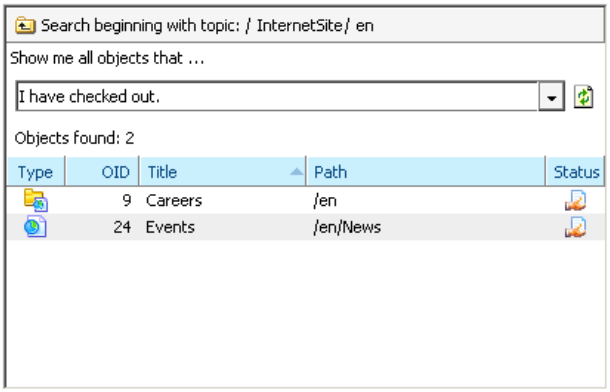


Fig. 28 – The "My objects" view

The objects are displayed in an object list. For each object, the object type, OID, title, path to the root object, and status are displayed. By clicking on the header row of a column, you can sort the object list according to the criterion in the column. Another click reverses the sort order. You can mark one or more objects in the list in order to edit them (see chapter 12 "Finding Objects and Editing Them Together").

The "My worklist" view

Choose *View* → *My worklist* to open the "My worklist" view. This view is personalized as it considers your assignments to workflow activities. Only those objects that are assigned a workflow and that you may edit are displayed. All the workflows and activities for which exist objects that you may currently edit are available as filters. The versions of the workflow objects are also considered.

For detailed information on the workflow of Livelink WCM Server, refer to chapter 8 "Content Workflow".

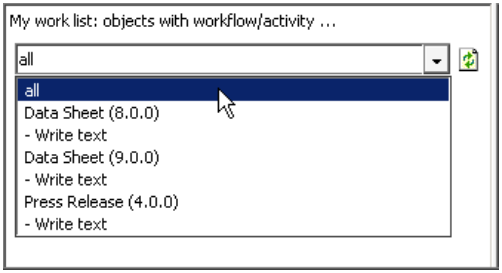


Fig. 29 – Filters in the "My worklist" view

After selecting a filter, all objects that are assigned to the selected workflow or that are in the selected activity are displayed. Select the entry *all* to display all objects that are in an activity you are assigned to.

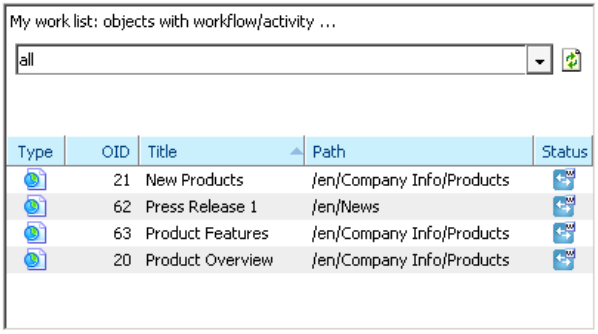


Fig. 30 – The "My worklist" view

The objects are displayed in an object list. For each object, the object type, OID, title, path to the root object, and status are displayed. By clicking on the header row of a column, you can sort the object list according to the criterion in the column. Another click reverses the sort order. You can mark an object in the list in order to edit it.

The “Search” view

Note: The search functionality is only available if you are assigned to the functional area “Search” and if at least one Search server is installed in your WCM system. In the Configuration dialog box, you can determine which Search server is to be used for searching. Moreover, an internal collection must exist for the website that is to be searched. For information on collections, refer to the online help of the Admin client.

To perform a search:

1. Choose *View* → *Search*.

   In the navigation area, the Search dialog box opens.

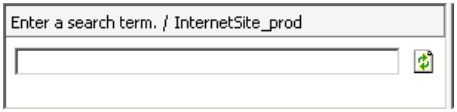


Fig. 31 – The “Search” view

2. Enter your search term in the field.
3. Click the *Search* icon.

   Icon for starting the search

   The search result is displayed in an object list.

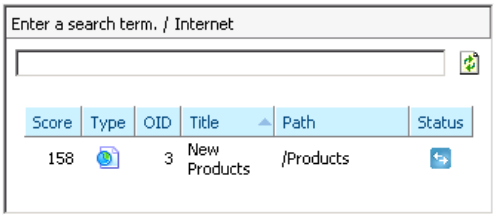


Fig. 32 – Result of a search

For each object, the score, object type, OID, title, and status are displayed. The score indicates the significance of the hit. A document containing the search term in the title gets a higher score than a document in which the search term was found in the text.

By clicking on the header row of a column, you can sort the object list according to the criterion in the column. Another click reverses the sort order. You can mark one or more objects in the list in order to edit them.

The Action List

Time-consuming operations (import, move and copy, delete topics, edit objects by means of the object filter) are performed asynchronously. Directly after starting such an operation, you can continue with your work, while the operation is being performed in the background.

When an asynchronous operation has been started, it is added to the so-called *action list*. By means of the action list, you can watch the progress of your asynchronous actions. The action list is displayed in a separate window.

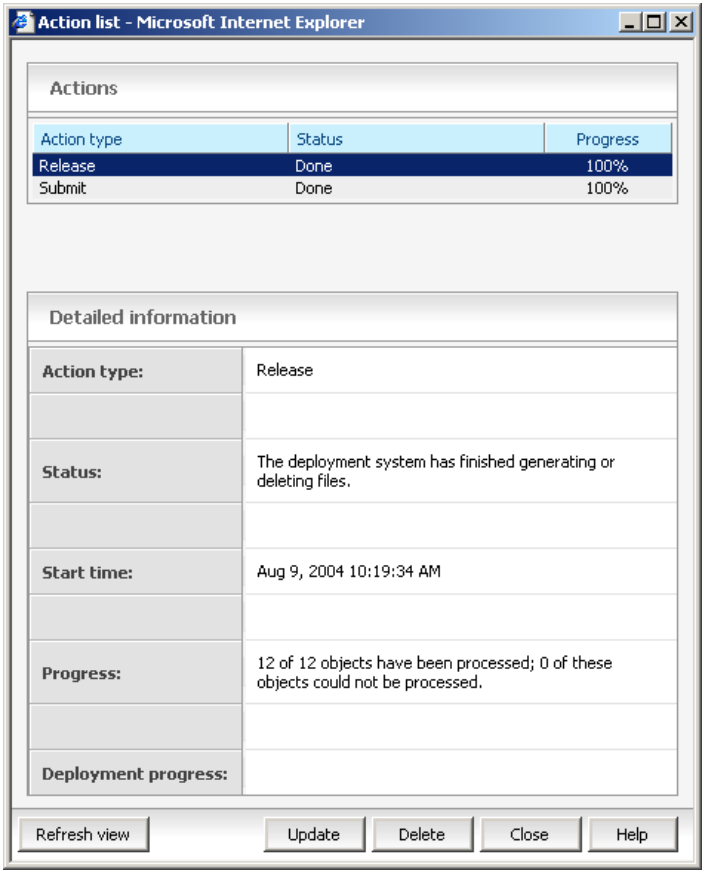


Fig. 33 – The action list

The following functions are available for the action list:

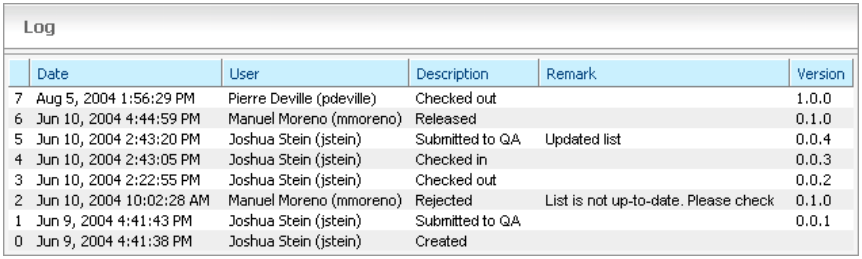
- *Refresh view:* refreshes the navigation area of the Content client after a time-consuming action
- *Update:* refreshes the action list
- *Delete:* removes the marked entry from the action list

> **Note:** As soon as you exit the Content client, all entries in the action list are deleted automatically.

The Log

> **Note:** For viewing the Log dialog box, you require the functional area "Log dialog". To check your functional areas, choose *Tools* → *Login info*.

The log is the history of the current object. It contains all actions that have been performed on the object. Choose *View* → *Log* to open the Log dialog box.


Log

	Date	User	Description	Remark	Version
7	Aug 5, 2004 1:56:29 PM	Pierre Deville (pdeville)	Checked out		1.0.0
6	Jun 10, 2004 4:44:59 PM	Manuel Moreno (mmoreno)	Released		0.1.0
5	Jun 10, 2004 2:43:20 PM	Joshua Stein (jstein)	Submitted to QA	Updated list	0.0.4
4	Jun 10, 2004 2:43:05 PM	Joshua Stein (jstein)	Checked in		0.0.3
3	Jun 10, 2004 2:22:55 PM	Joshua Stein (jstein)	Checked out		0.0.2
2	Jun 10, 2004 10:02:28 AM	Manuel Moreno (mmoreno)	Rejected	List is not up-to-date. Please check	0.1.0
1	Jun 9, 2004 4:41:43 PM	Joshua Stein (jstein)	Submitted to QA		0.0.1
0	Jun 9, 2004 4:41:38 PM	Joshua Stein (jstein)	Created		

Fig. 34 – The Log dialog box

For each action, the date (with time), responsible user, description, version, and – if existing – a remark are displayed. By clicking on the header row of a column, you can sort the log entries according to the criterion in the column. For example, clicking on *Version* sorts the list in ascending chronological order. To reverse the sort order, click the header row again. This way you can clearly arrange long lists.

Available functions

The functions in the Log dialog box are called via the context menu. The following functions are available if you mark a log entry and right-click it.

- *Display version:* opens a new window in which the content and some metadata of the marked object version are displayed

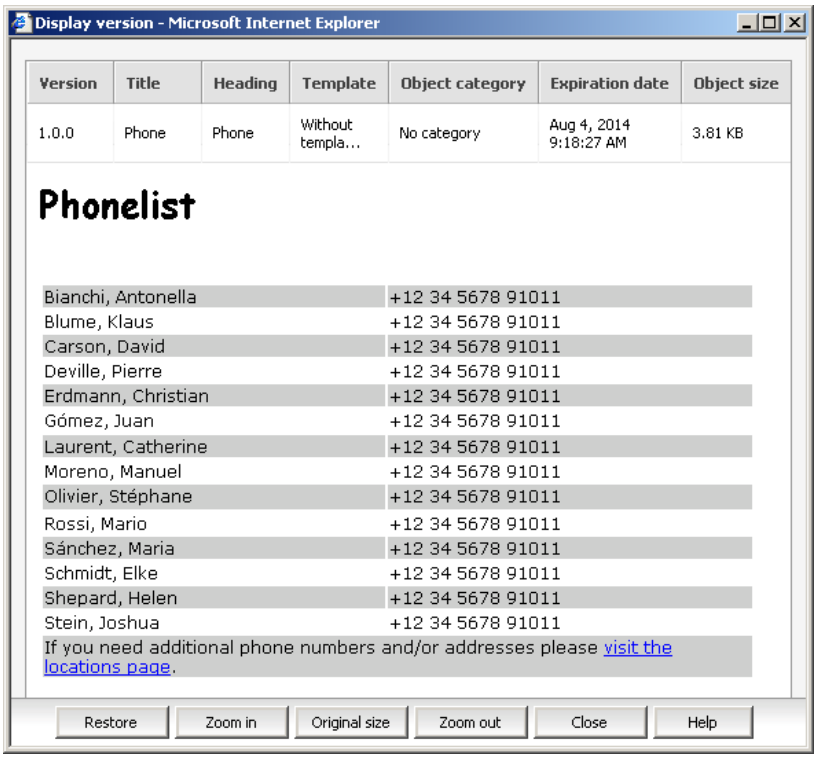


Fig. 35 – "Display version" function in the Log dialog box

- *Compare versions* (This function is only active if two log entries are marked.): opens a new window in which the two marked versions of the object are displayed one below the other. This makes it easy to compare the two object versions.

- *Compare with current version:* opens a new window in which the current version and the marked version of the object are displayed one below the other. This makes it easy to compare the two object versions.
- *Restore old version:* replaces the current version of the object with the marked version
- *Add new remark:* opens a dialog box in which you can enter information on the marked object version

Help on the Content client

When working in the Content client, you are assisted by wizards and online help.

Wizards

A wizard starts automatically when you choose a function, such as adding an object, and guides you through the process step by step.

Online help

The online help of the Content client is implemented as a context-sensitive web help. The help window for the relevant topic is opened depending on the dialog box you are currently working in. You can call the help by clicking the *Help* button or – when using Internet Explorer – by pressing the F1 key.

When you choose *Help topics* on the *Help* menu, the start page of the help opens regardless of the current context.

CHAPTER 4

Configuring the Content Client

Livelink WCM Server offers you several possibilities to configure the layout of the Content client and the procedures for editing objects. In this way, you can tailor the client to suit your needs. The settings you make are saved as a so-called *profile* and can be activated at any time.

By means of the Content client, you can create several user-defined profiles. For example, if you work at different work stations and have assigned editor programs to the different file types, it may be useful to create a personalized profile for each working environment.

The profile you used last is saved in a cookie in the browser. If your browser accepts persistent cookies, the profile saved in the cookie is used automatically the next time you log in to the WCM system.

The entire configuration of the Content client and the administration of the profiles are performed in the Configuration dialog box. Choose *Tools* → *Configuration* to open this dialog box.

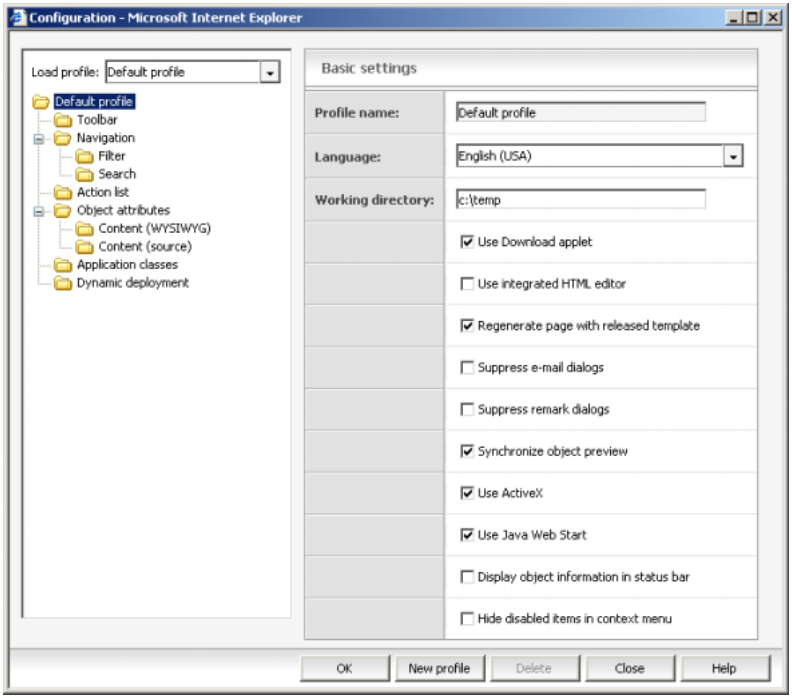


Fig. 36 – The Configuration dialog box

The topmost node of the tree structure in the left window pane is the current profile. The subordinate elements are the elements that can be configured. The individual settings of an element are displayed in the right window pane and can be edited there.

You can make settings for the following areas:

- basic settings of the profile, see section “Configuring the Basic Settings” starting on page 110
- toolbar of the Content client, see section “Configuring the Toolbar” on page 116

- navigation, see section “Configuring the Navigation” starting on page 117
- action list, see section “Configuring the Action List” starting on page 119
- object-dependent attributes, see section “Configuring Object Attributes” starting on page 120
- application classes, see section “Configuring Application Classes” starting on page 121
- dynamic deployment, see section “Configuring the Dynamic Deployment” on page 125

This chapter describes how to work with profiles. It also summarizes the different possibilities to tailor the Content client to your needs.

Adding a Profile

To add a profile:

1. Choose *Tools* → *Configuration* to open the Configuration dialog box.
2. Click the *New profile* button.

   In the *Profile name* field, the entry *New profile* is displayed.
3. Overwrite the entry *New profile* with the name of your new profile.
4. Do one of the following:
   - Click the *OK* button to save your settings and close the Configuration dialog box.
   - For each profile, you can specify settings in multiple configuration sections. To make further settings, select the appropriate configuration item in the tree on the left.

Configuring the Basic Settings

To specify the basic settings of the profile:

1. Choose *Tools* → *Configuration* to open the Configuration dialog box.
2. In the left window pane of the Configuration dialog box, mark the topmost node of the tree structure. This is the name of the profile.

   The basic settings are displayed in the right window pane.

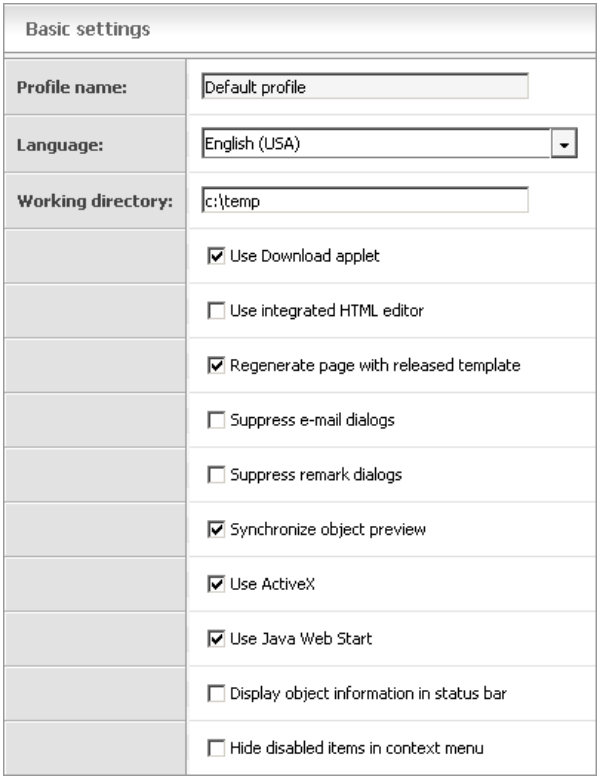


Fig. 37 – Configuring the basic settings of a profile

The following table gives an overview of the individual configuration options.

Table 10 – Basic settings of a profile

Basic settings	Effect if selected
Profile name	Name of the current profile. You cannot change the name of the default profile.
Language	Language of the graphical user interface of the Content client. This setting takes priority over the language setting made for your user ID in the Admin client.
Working directory	Local directory. The Download applet saves the files of the checked-out and edited objects in this directory.
Use Download applet	Use Download applet to check objects out and in See section “The Download Applet” starting on page 113
Use integrated HTML editor	Use integrated HTML editor to edit HTML objects See section “The Integrated HTML Editor” starting on page 114
Regenerate page with released template	The function “Regenerate page” uses the released template, even in the Edit view. **Note:** This setting only applies to static deployment system. For information on the configuration options for dynamic deployment systems, refer to section “Configuring the Dynamic Deployment” on page 125.

Basic settings	Effect if selected
Suppress E-mail dialogs	Never display an E-mail dialog box after performing a staging step. The person responsible for the next work step does **not** get a message. You can also make this setting directly in the E-mail dialog box. See section “Sending E-Mails” starting on page 92
Suppress remark dialogs	Never display a Remark dialog box after adding objects or editing an object content. You can also make this setting directly in the Remark dialog box.
Synchronize object preview	The object preview is synchronized with the navigation of the Content client. The object currently selected in the navigation area is displayed in the object preview of the browser window.
Use ActiveX	Allow the use of ActiveX controls. For security reasons, it may be recommendable to clear this check box. This may slow down execution of certain functions.
Use Java Web Start	Activate Java Web Start. This is required for starting the workflow modeler automatically when checking out objects of type “Workflow”. Additionally, the check box *Use Download applet* must be selected.
Display object information in status bar	Display object information of the current object in the left area of the status bar
Hide disabled items in context menu	Do not show disabled items in the context menu **Note:** If you use the Mozilla browser and select this check box, there may be display problems in the InSite Editing context menu. However, this does not restrict the functionality.

3. Make the desired settings.
4. Do one of the following:
   - Click the *OK* button to save your settings and close the Configuration dialog box.
   - To specify further settings for the profile, select another configuration section in the tree on the left.

In the following sections, the individual basic settings are described in more detail.

The Download Applet

An applet is a small program that is loaded from the Internet by the browser and is run on the recipient's computer.

The Download applet of the Content client assists you with all download processes (from the server to the local hard disk) and upload processes (from the local hard disk to the server). To edit an object, you have to check it out first. By doing so, you load the file that represents the object from the server to your local hard disk (download). When you have finished editing the object, you check it in again. This transfers the edited file from your local hard disk to the server (upload).

Use Download applet check box cleared

> **Note:** For performance and usability reasons, we strongly recommend that you **always** work with the Download applet.

For security reasons, it may be necessary to prevent access by the browser to your local hard disk. This is ensured by clearing the *Use Download applet* check box. If you clear this check box, you will have to perform all download and upload operations manually. For the exact procedures, refer to sections "Checking out without Download applet" on page 171 and "Checking in without Download applet" on page 176.

The advantage of working without the Download applet is that you have full control of the data transfer. This guarantees maximum security. The disadvantage is that working becomes relatively inconvenient.

***Use Download applet* check box selected**

You can avoid manual downloading and uploading of the file by selecting the *Use Download applet* check box.

For the exact procedures, refer to sections “Checking out with Download applet” on page 169 and “Checking in with Download applet” on page 174.

> **Important:** Do not change the working directory or the setting of the *Use Download applet* check box while working with the Content client. Otherwise problems may occur as the system cannot calculate the file name and file path for automatic upload and download.

The Integrated HTML Editor

The Configuration dialog box provides – especially for editing HTML files – the check box *Use integrated HTML editor*. The integrated HTML editor is an applet and thus runs within the browser.

When you start the integrated HTML editor for the first time, a security warning is displayed. You are asked to accept the certificate for the editor.

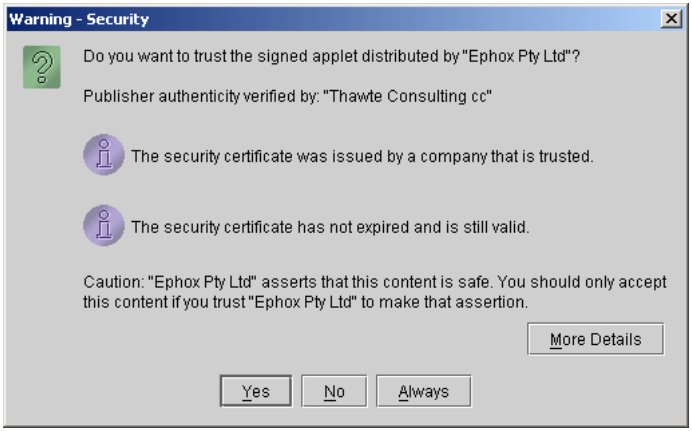

Warning - Security
Do you want to trust the signed applet distributed by "Ephox Pty Ltd"?
Publisher authenticity verified by: "Thawte Consulting cc"
The security certificate was issued by a company that is trusted.
The security certificate has not expired and is still valid.
Caution: "Ephox Pty Ltd" asserts that this content is safe. You should only accept this content if you trust "Ephox Pty Ltd" to make that assertion.
More Details
Yes
No
Always

Fig. 38 – Security warning when starting the integrated HTML editor

You have the following options:

- *Yes:* Click this button to confirm the security prompt. This is necessary for working with the integrated HTML editor.
- *No:* If you do not confirm the prompt, you cannot work with the integrated HTML editor.
- *Always:* Click this button to confirm the security prompt. After that, this security prompt no longer appears when you are working with the integrated HTML editor.
- *More Details:* Click this button to request more information about the certificate.

For information on the exact procedure when using the editor, refer to sections “Checking out with integrated HTML editor” on page 172 and “Checking in with integrated HTML editor” on page 176.

Configuring the Toolbar

To specify the settings for the toolbar:

1. Choose *Tools* → *Configuration* to open the Configuration dialog box.
2. Mark the tree element *Toolbar* in the left window pane of the Configuration dialog box.

   The possible settings are displayed in the right window pane.

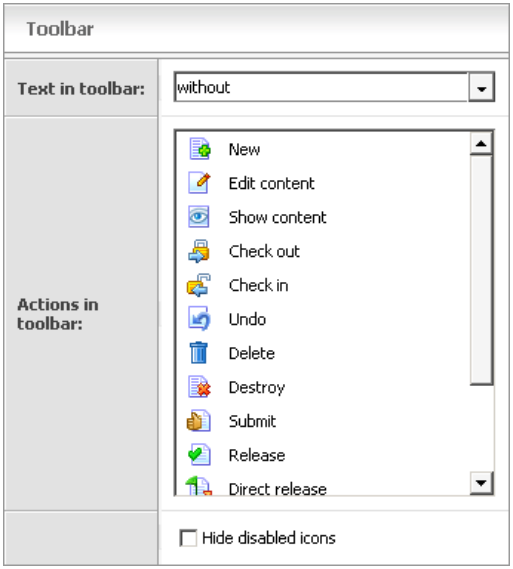


Fig. 39 – Configuring the toolbar

3. Make the desired settings:
   - *Text in toolbar:* Specify whether you want text to be displayed on the toolbar icons and where this text is located.
   - *Actions in toolbar:* Specify which functions are to be represented by toolbar icons.

- *Hide disabled icons:* Select this check box if you do not want to display deactivated icons.

  Note: If you use the Mozilla browser and selected *below* for *Text in toolbar*, you must not select this check box.

4. Do one of the following:
   - Click the *OK* button to save your settings and close the Configuration dialog box.
   - To specify further settings for the profile, select another configuration section in the tree on the left.

Configuring the Navigation

To specify the settings for the navigation:

1. Choose *Tools* → *Configuration* to open the Configuration dialog box.
2. Mark the tree element *Navigation* or one of its sub-elements in the left window pane of the Configuration dialog box.

   The possible settings are displayed in the right window pane.

   The following table describes for which areas of the navigation you can make settings. For detailed information, refer to the online help of the Content client.

Table 11 – Navigation settings of a profile

Item	Possible settings
Navigation	Layout of the object tree nodes Maximum number of objects displayed below a topic or template Maximum number of objects displayed in the “Object list” view Root object for navigation Selection whether the heading is displayed in the object tree (in addition to the title)
Filter	Maximum number of objects displayed in a filter hitlist Maximum number of objects found by the filter Selection of filters available in the “Object filter” and “My filter” views
Search	Maximum number of objects displayed in the hitlist of the “Search” view Selection of the Search server to be used for the search

3. Make the desired settings.
4. Do one of the following:
   - Click the *OK* button to save your settings and close the Configuration dialog box.
   - To specify further settings for the profile, select another configuration section in the tree on the left.

Configuring the Action List

For general information on the action list, refer to section “The Action List” starting on page 101.

To specify the settings for the action list:

1. Choose *Tools* → *Configuration* to open the Configuration dialog box.
2. Mark the tree element *Action list* in the left window pane of the Configuration dialog box.

   The possible settings are displayed in the right window pane.

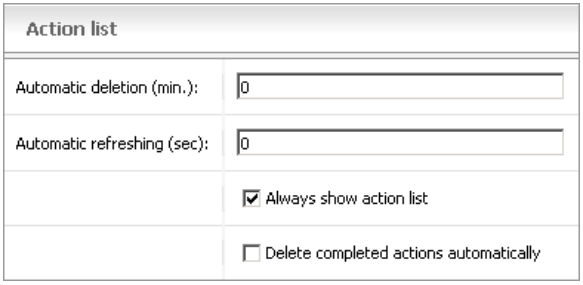


Fig. 40 – Configuring the action list

3. Make the desired settings:
   - *Automatic deletion (min.):* In this field, you can enter after how many minutes completed actions are to be deleted from the action list.
   - *Automatic refreshing (sec):* In this field, you can enter after how many seconds the action list is to be refreshed automatically. If you enter “0”, the action list is not updated automatically.
   - *Always show action list:* When this check box is selected, the action list window opens automatically when an asynchronous operation is started.

If you clear the check box, the action list is not displayed automatically when an asynchronous action is started. By choosing *Tools* → *Action list*, you can open the action list manually at any time.

- *Delete completed actions automatically:* If you select this check box, entries for finished asynchronous actions will be deleted automatically from the action list.

4. Do one of the following:
   - Click the *OK* button to save your settings and close the Configuration dialog box.
   - To specify further settings for the profile, select another configuration section in the tree on the left.

Configuring Object Attributes

To specify the settings for the object attributes:

1. Choose *Tools* → *Configuration* to open the Configuration dialog box.
2. Mark the tree element *Object attributes* or one of its sub-elements in the left window pane of the Configuration dialog box.

   The possible settings are displayed in the right window pane.

   The following table describes the object attributes you can make settings for. For detailed information, refer to the online help of the Content client.

Table 12 – Settings for the object attributes in a profile

Item	Possible settings
Object attributes	Maximum number of entries to be displayed in the References dialog box Maximum number of entries to be displayed in the Log dialog box Maximum number of objects that can be displayed in the right window pane of the Content client via *View → Subordinate objects*
Content (WYSIWYG)	Specifies the object types that can be displayed in the right window pane of the Content client via *View → Content (WYSIWYG)*
Content (source)	Specifies the object types that can be displayed in the right window pane of the Content client via *View → Content (source)*

3. Make the desired settings.
4. Do one of the following:
   - Click the *OK* button to save your settings and close the Configuration dialog box.
   - To specify further settings for the profile, select another configuration section in the tree on the left.

Configuring Application Classes

If you use the Download applet for editing objects, the object you check out is automatically opened in the associated editor. For this purpose, you must assign the editors, so-called *applications*, to the different file types. The assignment between an application (e.g. Word) and any number of file types is called *application class*. The file types are represented by their respective extension (e.g. **doc** for the file type “Word document”).

To manage application classes, open the Configuration dialog box via *Tools* → *Configuration* and select the tree element *Application classes* in the left window pane. In the right window pane, you can add, edit, and delete application classes.

Adding an Application Class

To add an application class:

1. Mark the tree element *Application classes* in the left window pane of the Configuration dialog box.
2. Click the *New application class* button in the right window pane.

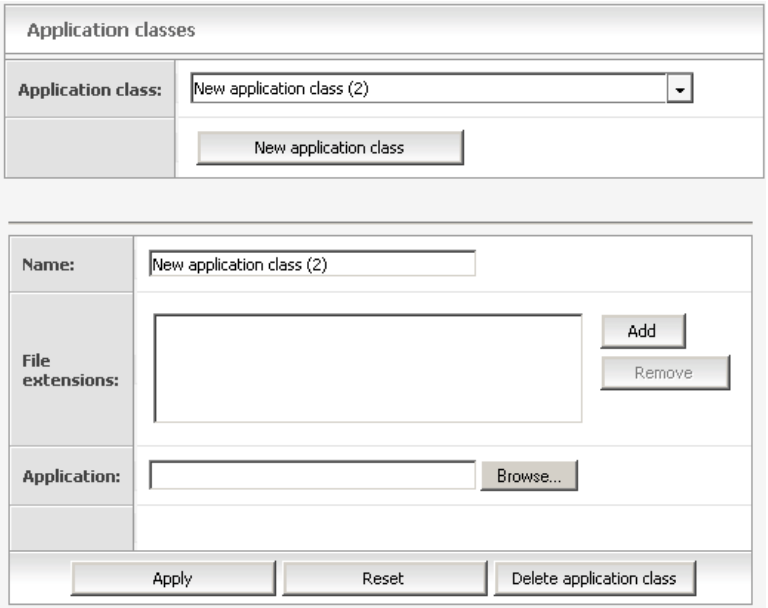


Fig. 41 – Adding an application class

3. In the *Name* field, enter a name for the application class.
4. Click the *Add* button.

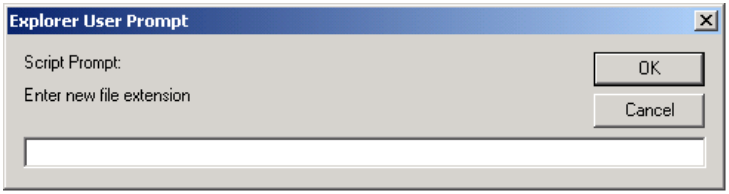


Fig. 42 – Entering the file extension for an application class

5. Enter the extension of the file type to which you want to assign an application and click the *OK* button.

   The file extension is added to the list. You can repeat steps 4 and 5 for all file extensions that you want to assign to the same application.

   > **Note:** It does not make sense to use a file extension in more than one application class, i.e. assign more than one application to the file type. This would make it impossible to define which editor is to be opened when the object is checked out.

6. Use the *Browse* button to select the desired editor program.

   The full path to the program is displayed in the *Application* field.

7. Click the *Apply* button.

8. Do one of the following:

   - Click the *OK* button to save your settings and close the Configuration dialog box.
   - To specify further settings for the profile, select another configuration section in the tree on the left.

Editing an Application Class

To edit an application class:

1. Mark the tree element *Application classes* in the left window pane of the Configuration dialog box.
2. In the *Application class* drop-down list in the right window pane, click the desired application class.

   The settings of the selected application class are displayed. All settings can be changed.
3. Make the desired changes.
4. Click the *Apply* button.
5. Do one of the following:
   - Click the *OK* button to save your settings and close the Configuration dialog box.
   - To specify further settings for the profile, select another configuration section in the tree on the left.

Deleting an Application Class

To delete an application class:

1. Mark the tree element *Application classes* in the left window pane of the Configuration dialog box.
2. In the *Application class* drop-down list in the right window pane, click the desired application class.
3. Click the *Delete application class* button.
4. Do one of the following:

- Click the *OK* button to save your settings and close the Configuration dialog box.
- To specify further settings for the profile, select another configuration section in the tree on the left.

Configuring the Dynamic Deployment

By means of the configuration for the dynamic deployment, you can determine the object view that is displayed when you select a dynamic deployment system in the Content client.

To specify the settings for the dynamic deployment:

1. Choose *Tools* → *Configuration* to open the Configuration dialog box.
2. Mark the tree element *Dynamic deployment* in the left window pane of the Configuration dialog box.

   The possible settings are displayed in the right window pane.

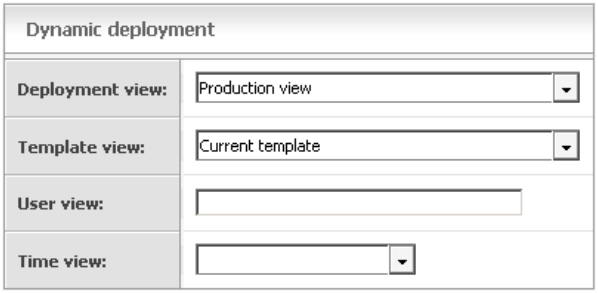


Fig. 43 – Configuring the dynamic deployment

3. Make the desired settings:

- *Deployment view:* specifies whether the Edit, QA, or Production view of the current object is displayed in the object preview.

  If you do not make any settings here, the view corresponding to the type of the dynamic deployment system is displayed by default. If the deployment system is a Production deployment system, for example, the Production view is shown.

- *Template view:* specifies whether the current version of the template in the Edit view or the released version of the template is used for generating the HTML page.

  If you do not make any settings here, the released version of the template is used by default.

- *User view:* makes it possible to display objects from the point of view of a certain user. The current version of the objects in the Production view is displayed, including all changes the respective user has made. Also displayed are all changes that other users made before the object was edited by the selected user. Enter the ID of the desired user in the field.

  Note: Which changes are displayed also depends on the selected deployment view. For example, the changes of the selected user are only visible in the QA view after the object has been submitted.

- *Time view:* makes it possible to display the status the objects had at a certain point in time. Enter a date (with time) in the field. Click the arrow icon to open a calendar for selecting the desired date.

> **Note:** The respective pages are generated from the source code of the objects at the selected point in time. If this includes reading the content or metadata of other WCM objects, the current version of the respective WCM objects is used. As a result, the time view generated by the dynamic deployment may – especially in connection with JSP pages – not correspond to the actual status of the website at the selected point in time.

4. Do one of the following:
   - Click the *OK* button to save your settings and close the Configuration dialog box.
   - To specify further settings for the profile, select another configuration section in the tree on the left.

Using a Profile

To use a saved profile:

1. Choose *Tools* → *Configuration* to open the Configuration dialog box.
2. In the *Load profile* drop-down list, click the profile that you want to use.

   The selected profile is loaded to the Configuration dialog box.
3. Click the *OK* button.

Editing a Profile

To edit a saved profile:

1. Choose *Tools* → *Configuration* to open the Configuration dialog box.
2. In the *Load profile* drop-down list, click the profile that you want to edit.

   The selected profile is loaded to the Configuration dialog box.
3. Make the desired changes.
4. Click the *OK* button.

Deleting a Profile

To delete a stored profile:

1. Choose *Tools* → *Configuration* to open the Configuration dialog box.
2. In the *Load profile* drop-down list, click the profile that you want to delete.

   Note: You cannot delete the default profile.

   The selected profile is loaded to the Configuration dialog box.
3. Click the *Delete* button.

   The profile is removed from the *Load profile* drop-down list.
4. Click the *OK* button.

Customizing the Content Client

Livelink WCM Server offers you various possibilities to customize the Content client to your individual needs.

Basically, you can use customizing options to hide certain user interface elements. This makes working with the Content client even easier, especially for users with a clearly defined working environment. You have the following possibilities to tailor the Content client to your needs:

- through access rights, see the following section
- through functional areas, see section “Customizing Through Functional Areas” on page 130
- through profiles, see section “Customizing Through Profiles” on page 130

Customizing Through Access Rights

For general information on access rights, refer to section “Access Rights” on page 50.

The access rights determine how the users may access the objects, e.g. whether they may only read the object or whether they are also allowed to change it. Certain icons and menu items are, for example, disabled if the user does not have the required right.

In the Admin client, the access rights can be preset for each principal. These settings are called default object rights. If one of these principals is added to the access control list, these settings are used as the initial settings for the access rights. For more information, refer to the Livelink WCM Server Administrator Manual.

With the Content client, you modify the access rights for each object at any time.

Customizing Through Functional Areas

For general information on functional areas, refer to section “Functional Areas” on page 61.

Functional areas determine which types of objects a user may add, check out, and check in. The functional areas such as “Basic” are by default linked with object types. Only users that have the corresponding functional area can add and edit objects of this object type. When adding an object, users may only select the object types for which they have the required functional areas.

In addition, functional areas determine which views and dialog boxes are available to the user. For example, the Access rights dialog box is only displayed if the user has the functional area “Access rights dialog”.

Functional areas are assigned to users, groups, and roles via the Admin client. Here, the WCM administrator can also add functional areas and edit existing ones. For more information, refer to the Livelink WCM Server Administrator Manual.

Customizing Through Profiles

Profiles provide the following customizing options:

- Toolbar: You can determine which functions are to be displayed as icons in the toolbar, whether text is to be shown for these icons, and where this text is to be positioned. Moreover, you can hide deactivated icons (see section “Configuring the Toolbar” on page 116).
- Context menu: You can hide deactivated items from the context menu (see section “Configuring the Basic Settings” on page 110).
- Navigation: You can choose from five different icons for the nodes in the object tree (see section “Configuring the Navigation” on page 117).

- Filter selection: You can determine which filters are to be available in the “Object filter” and “My filter” views. This way, you can restrict the selection to the filters that are most frequently used (see section “Configuring the Navigation” on page 117).
- Remark and e-mail dialog boxes: The Content client can be configured in such a way that E-mail and Remark dialog boxes are never displayed (see section “Configuring the Basic Settings” on page 110).
- Application classes: You can assign editors to the various file types, which will start automatically when an object of this type is checked out. This enables you to use your familiar authoring environment (see section “Configuring Application Classes” on page 121).

CHAPTER 5
Adding Objects

This chapter explains how to add objects to your website. The Content client offers you different possibilities of adding an object:

- the function “New → from object type”, see the following section
- the function “New → assembled object”
- the function “New → from template”, see section “Adding Objects on the Basis of a Template” on page 148
- the function “New → Relator”, see section “Adding Relators” on page 340
- the function “Import → linked objects”, see section “Importing Linked Objects” starting on page 154
- the function “Import → from ZIP file”, see section “Importing Objects from a ZIP File” starting on page 158
- the function “Copy”, see section “Copying an Object” starting on page 162

Adding Objects on the Basis of an Object Type

Supported actions

The wizard for adding objects on the basis of an object type helps you perform the following actions:

- add an object that does not yet exist as a file

  You can add an object that neither exists as a file on your local hard disk nor in the network. This is the case, for example, if you want to add an object of type “Topic” as an additional structural element in your website. Or you may want to add an empty page that will initially serve as a placeholder and will not be filled with content until later.

- add an object from an existing file

  You can add a WCM object based on a file (e.g. an HTML page or a graphic) which is stored on your local hard disk or in the network. The external file is imported into the WCM system. Under certain circumstances, objects referenced by this file are also imported.

- add an object with the direct release right

  You can add objects that do not have to be submitted to Quality Assurance, but can be released directly by the responsible editor. The objects must only be submitted to Quality Assurance for initial release.

Required rights and functional areas

You need the following rights and functional areas to add an object by means of the wizard:

- the access rights “Read” and “Create” for the topic to which the object is to be added

- the functional area linked with the object type of the new object. To check your functional areas, choose *Tools* → *Login info*.

The following sections describe how to add an object by means of the wizard.

Starting the Wizard

To start the wizard:

1. Change to the Edit view of the website.
2. In the navigation area, mark the topic to which you want to add the object.
3. Choose *Object* → *New* → *from object type* or click the appropriate icon.

   Icon for adding an object on the basis of an object type

   The wizard starts.
4. Follow the instructions of the wizard that guides you through the process of adding the object.

Specifying Title, Object Type, and Workflow

In the first dialog box, you specify the title and object type of the new object. If you want to add an object on the basis of an existing file, select the desired file. You can also assign a workflow to the object.

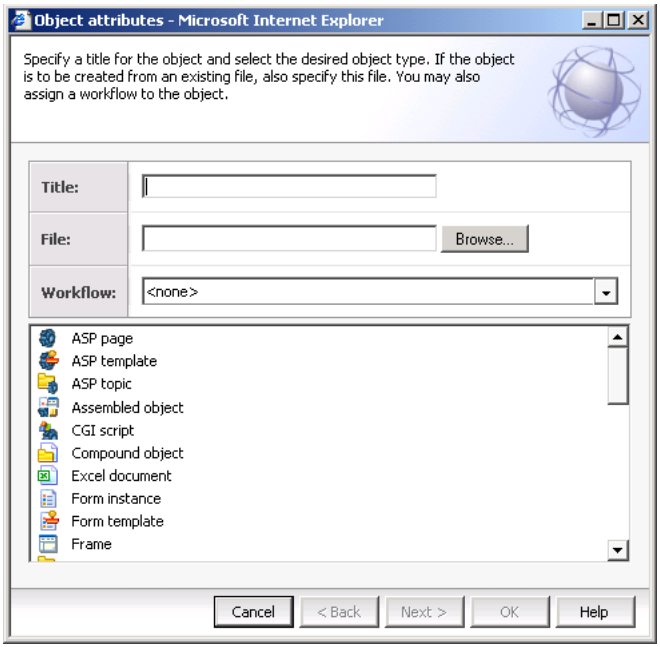


Fig. 44 – Specifying title, type, and workflow of a new object

- *Title:* Enter a name for the object. The title is used to identify the object and is displayed both in the header and in the navigation area.

  Entering a title is mandatory.

  > **Note:** Titles must not be more than 255 characters long. Also, you should adapt the length of the title in such a way that the resulting file name meets the requirements of your operating system. Under Windows, the absolute path to a topic and a generated page may not be longer than 240 and 260 characters respectively.

- *File:* If you want to use an existing file as the basis for the new object, click the *Browse* button to select the file.

> **Note:** Please choose only files that have a file extension.

After you have selected a file, the selection of object types will be restricted. Only those object types that match the file extension of the selected file are displayed.

- *Workflow:* If you want to assign a workflow to the new object, click the desired workflow in the drop-down list.
- Object type: Select the object type from the list, e.g. “HTML page”, “HTML template”, “Topic”

  Specifying an object type is mandatory. For some object types, it is possible to change the type subsequently.

Do one of the following:

- To confirm your entries and make further settings for the new object, click the *Next* button.
- Your entries might already be sufficient for adding the object. In this case, you can click the *OK* button to add the object.

Specifying Basic Properties

In this dialog box, you specify the basic properties of the object.

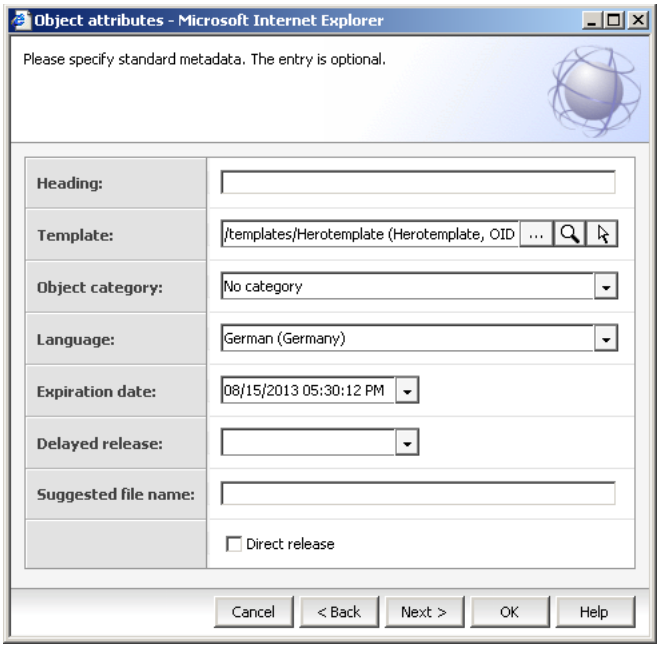


Fig. 45 – Specifying the basic properties of a new object

Note: If you do not make any entries in the *Template* and *Expiration date* fields, the settings of the parent topic are adopted automatically.

- *Heading:* Enter a heading for the object.
- *Template:* Assign a template to the object. By assigning a template, you define the layout of the page.

  Click the icon [...] to open a window with the template structure. Select the desired template in that window. If you know the OID of the template, you can enter it directly in the *Template* field.

  Click the icon [🔍] to view the selected template in the object preview. The icon [↖] takes you directly to the selected template.

- *Object category:* Use the drop-down list to assign an object category to the object. This enables you to categorize the object on the basis of its content.
- *Language:* Specify the language for the object. This item has no influence on the object content, but is used for classification purposes and as a search criterion in the object filter. Moreover, certain metadata, such as the date, are displayed in the selected language if they are integrated into the page generated for the object by means of placeholders, so-called WCM tags.
- *Expiration date:* Specify the expiration date (with time) for the object. The expiration date defines how long the object remains valid. The expiration date of the parent topic is used as the default setting.

  Click the arrow icon to open a calendar for selecting the desired date. The time can be set manually. When the object reaches the expiration date in the Edit view, a message is automatically sent to the address specified in the *E-mail Edit* field.
- *Delayed release:* Specify the publication date (with time) for the object. This determines when the object – after having been released by Quality Assurance – will be available for the end users in the Production view. Click the arrow icon to open a calendar for selecting the date. The time can be set manually.
- *Suggested file name:* Enter a name for the page which is generated from the WCM object. In many cases, the WCM system considers this suggestion.
- *Direct release:* You can only select this check box if the functional area “Direct release” has been assigned to you. If you select this check box, the object does not have to be submitted to Quality Assurance; it can be released directly by the responsible editor. The object must only be submitted to Quality Assurance for initial release.

Do one of the following:

- To confirm your entries and make further settings for the new object, click the *Next* button.
- Your entries might already be sufficient for adding the object. In this case, you can click the *OK* button to add the object.

Defining Object Responsibilities

In this dialog box, you define the object responsibilities. You can also enter the description, keywords, and the target group of the object.

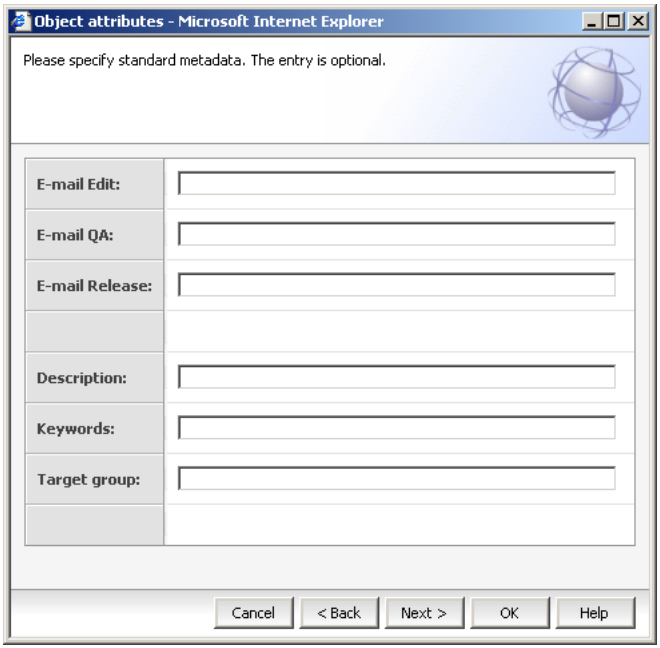


Fig. 46 – Specifying the responsibilities for a new object

Notes:

If you do not make any entries in the *E-mail Edit*, *E-mail QA*, *E-mail Release*, and *Target group* fields, the settings of the parent topic are adopted automatically.

Use a comma as separator between two e-mail addresses. If you want to enter the name of the user in addition to the e-mail address, write the name in quotation marks before the address and enclose the address in angle brackets.
Example: “Joshua Stein” <joshua.stein@company.example>, “Shepard, Helen” <helen.shepard@company.example>

- *E-mail Edit:* Enter the e-mail addresses of the people responsible for editing the object.

  If the Content client is configured accordingly, a message can be sent to these e-mail addresses as soon as the object has expired in the Edit view or has been rejected or destroyed by Quality Assurance.

- *E-mail QA:* Enter the e-mail addresses of the people responsible for quality assurance.

  If the Content client is configured accordingly, a message can be sent to these e-mail addresses as soon as the object has been submitted to Quality Assurance.

- *E-mail Release:* Enter the e-mail addresses to which a message can be sent – if the Content client is configured accordingly – when the object is released.

- *Description*, *Keywords*, *Target group:* Enter additional information on the object. You can use this information – like the other object information – as a search criterion in the object filter.

Notes:

In the *Keywords* field, commas are by default interpreted as separators between the individual entries. Double leading and trailing white spaces, such as spaces and tabs, are removed.

If an expression contains a comma that you do not want to be interpreted as a separator, enclose the expression in quotation marks. Example: “Paris, France”, French Revolution, 18th century

Do one of the following:

- To confirm your entries and make further settings for the new object, click the *Next* button.
- Your entries might already be sufficient for adding the object. In this case, you can click the *OK* button to add the object.

Entering Attribute Values

In this dialog box, you specify the values for the attributes of the object. The attributes of the object result from its association with an object type.

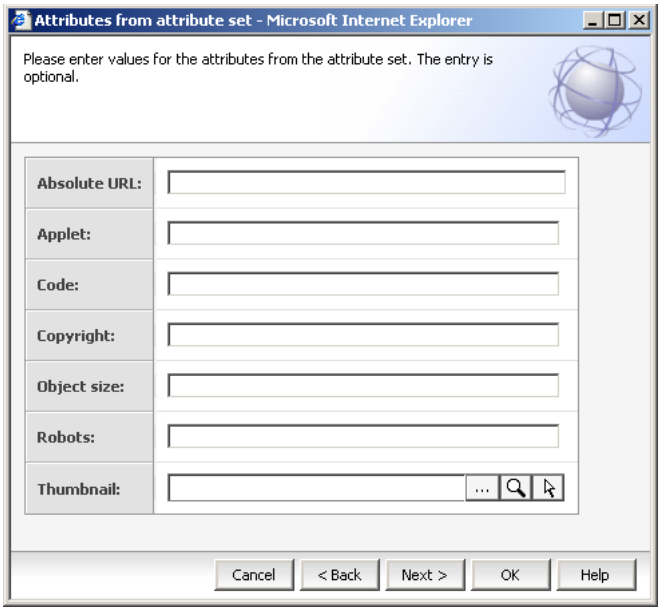


Fig. 47 – Specifying the attribute values for a new object

The fields of this dialog box are determined by the attribute set that is assigned to the object type. This assignment is variable and can be changed via *Tools* → *Website administration* → *Attribute sets* or via the Admin client. As an example, the attributes from the attribute set “html” are shown. By default, this attribute set is linked with the object type “HTML page”.

Do one of the following:

- To confirm your entries and make further settings for the new object, click the *Next* button.
- Your entries might already be sufficient for adding the object. In this case, you can click the *OK* button to add the object.

Entering Properties Values

> **Note:** This dialog box is only displayed if you have assigned an object category to the object.

In this dialog box, you specify the values for the properties of the object. The properties of the object result from its assignment to an object category.

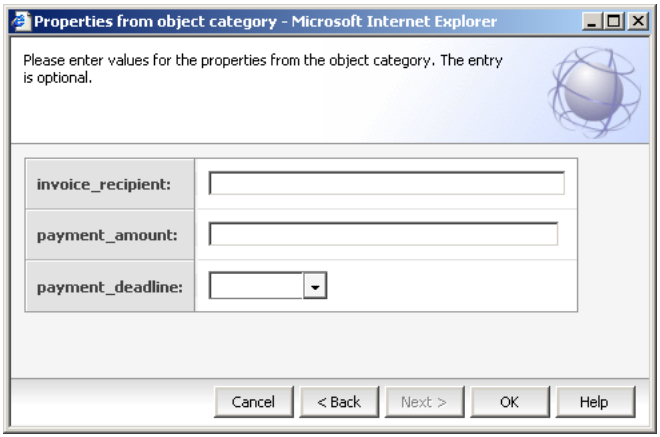


Fig. 48 – Specifying the property values for a new object

The fields of this dialog box are determined by the object category that is assigned to the object. The example shows the properties from the object category “Invoice”, which has been added via *Tools* → *Website administration* → *Object categories* or via the Admin client.

To add the object, click the *OK* button.

Selecting a Workflow Transition

> **Note:** This dialog box is only displayed if you have assigned a workflow to the object in one of the previous steps and if this workflow has more than one starting point or the starting point has more than one transition.

In this dialog box, you determine the first activity which is to be performed on the new object. Select one of the available transitions.

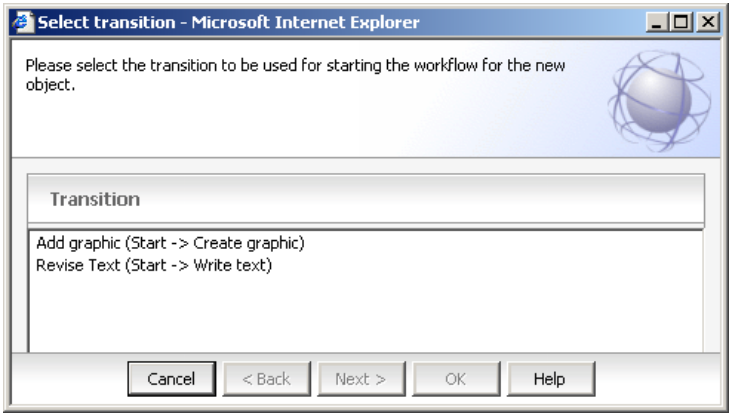

Select transition - Microsoft Internet Explorer
Please select the transition to be used for starting the workflow for the new object.
Transition
Add graphic (Start -> Create graphic)
Revise Text (Start -> Write text)
Cancel
< Back
Next >
OK
Help

Fig. 49 – Selecting a workflow transition

To add the object, click the *OK* button.

Entering a Remark

In the *Remark* field of this dialog box, you can enter information that is written to the log.

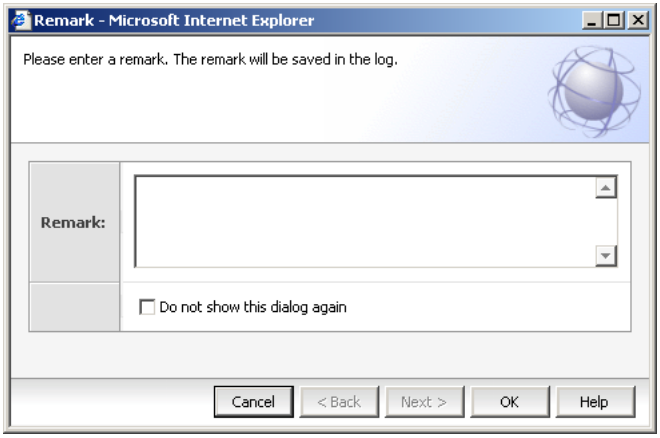


Fig. 50 – Entering a remark for a new object

Select the *Do not show this dialog again* check box if you do not want the Remark dialog box to be displayed after adding objects or editing object content. You can also make this setting in the Configuration dialog box.

To add the object, click the *OK* button.

Importing Embedded Objects

Embedded objects are objects referenced in an object. These objects can include images, CSS files, and JavaScript files contained in an HTML or JSP page. The following conditions must be fulfilled to import these objects together with the referencing object into the WCM system.

- the references are relative
- the embedded objects are located in the same directory as the referencing object or in a subdirectory
- the Download applet is used for working with the Content client

> **Note:** Only those objects are imported as embedded objects into the WCM system that conform to one of the object types “GIF image”, “JavaScript”, “JPEG image”, “PNG image”, or “Other”

If you click the *OK* button in the wizard for adding an object, the system checks whether the file specified in the second step contains embedded objects. If there are embedded objects, the following dialog box is displayed.

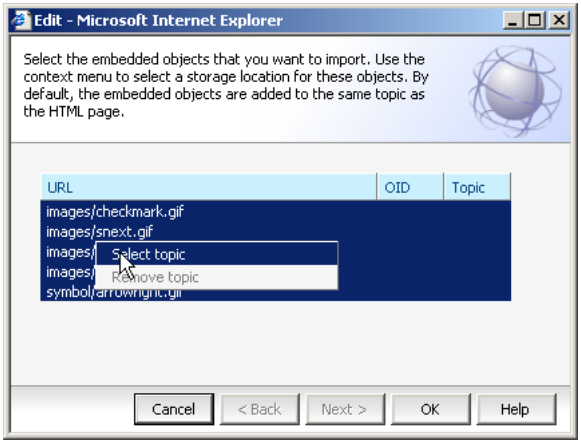

Edit - Microsoft Internet Explorer
Select the embedded objects that you want to import. Use the context menu to select a storage location for these objects. By default, the embedded objects are added to the same topic as the HTML page.
URL
OID
Topic
images/checkmark.gif
images/snext.gif
Select topic
Remove topic
Cancel
< Back
Next >
OK
Help

Fig. 51 – Selecting embedded objects when adding an object

In this dialog box, you determine which embedded objects are to be imported into the WCM system together with the referencing object. You can deselect individual – or all – objects. These objects will not be imported. If an object with the same name and object type already exists in your website, the *OID* column shows the OID of this object. Deselect existing objects to prevent adding multiple objects containing the same file, e.g. the same image.

By default, the embedded objects are added to the topic containing the referencing object. To specify a different storage location for individual objects, choose *Select topic* on the context menu and mark the desired topic in the selection dialog box. The *Topic* column will show the OID of the storage location.

To add the object, click the *OK* button.

The system creates the new object – together with the embedded objects – and displays them in the navigation area. The object's status is “changed”.

Adding Objects on the Basis of a Template

The wizard for adding an object on the basis of a template enables you to determine the layout of the new object already when adding the object. Thumbnails of the templates facilitate the search for the desired template.

> **Note:** You **cannot** use this wizard if you want to add an object based on an existing file.

Required rights and functional areas

You need the following rights and functional areas to add an object by means of the wizard:

- the access rights “Read” and “Create” for the topic to which the object is to be added
- the functional area linked with the object type of the new object. To check your functional areas, choose *Tools* → *Login info*.

The following sections describe how to add an object on the basis of a template.

Starting the Wizard

To start the wizard:

1. Change to the Edit view of the website.
2. In the navigation area, mark the topic to which you want to add the object.
3. Choose *Object* → *New* → *from template*.

   The wizard starts.
4. Follow the instructions of the wizard that guides you through the process of adding the object.

Specifying Title, Template, and Workflow

In the first dialog box, you determine the title of the new object and select a template. You can also assign a workflow to the object.

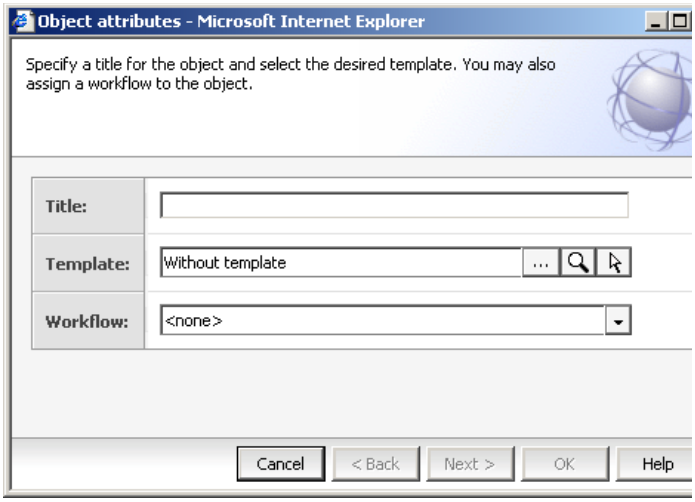


Fig. 52 – Specifying title and template when adding objects on the basis of a template

- *Title:* Enter a name for the object. The title is used to identify the object and is displayed both in the header and in the navigation area.

  Entering a title is mandatory.

  > **Note:** Titles must not be more than 255 characters long. Also, you should adapt the length of the title in such a way that the resulting file name meets the requirements of your operating system. Under Windows, the absolute path to a topic and a generated page may not be longer than 240 and 260 characters respectively.

- *Template:* Assign a template to the object. By assigning a template, you define the layout of the page.

Specifying a template is mandatory.

Click the icon [...] to open a window with the template structure. If configured (see section “Activating Thumbnails for Objects” on page 422), thumbnails for the individual templates will be displayed.

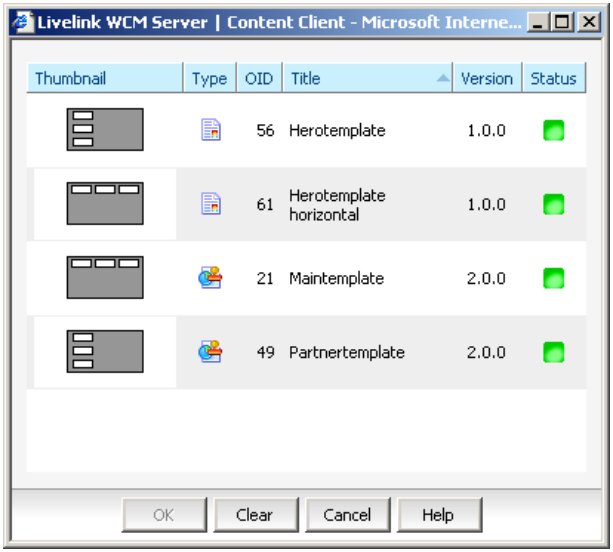


Fig. 53 – Template selection with thumbnails

Select the desired template and confirm by clicking the *OK* button. If you know the OID of the template, you can enter it directly in the *Template* field.

Click the icon [🔍] to view the selected template in the object preview. The icon [↖] takes you directly to the selected template.

- *Workflow:* If you want to assign a workflow to the new object, click the desired workflow in the drop-down list.

To confirm your entries and make further settings for the new object, click the *Next* button.

Specifying Object Type and Basic Properties

In this dialog box, you define the object type and basic properties for the new object.

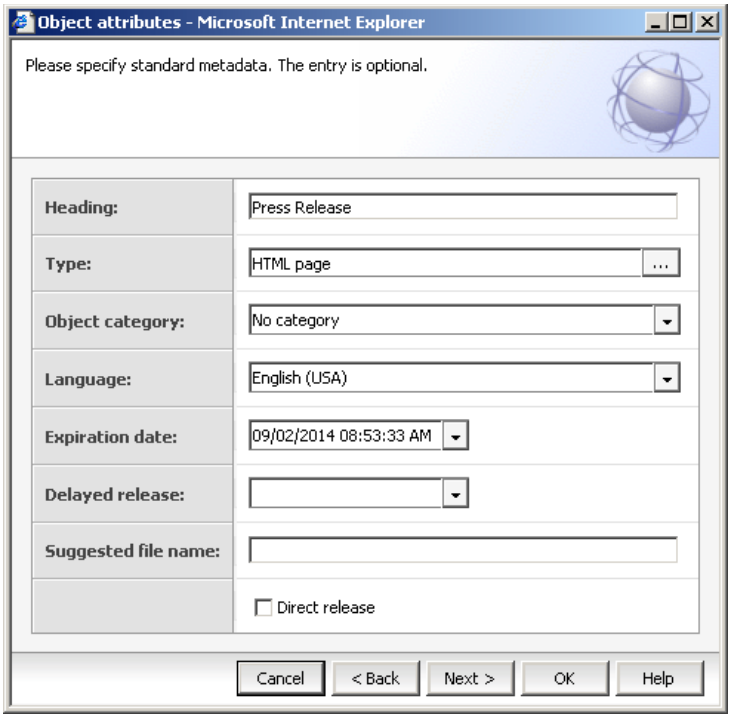


Fig. 54 – Specifying type and basic properties when adding objects on the basis of a template

- *Heading:* Enter a heading for the object.
- *Type:* Determine the type of the object. Click the icon [...] to open a dialog box with a list of all object types
- *Object category:* Use the drop-down list to assign an object category to the object. By assigning an object category, you can categorize the object based on its contents or create an assembled object

based on this category. If an object category is assigned to the template selected in the previous step, this category is automatically selected here.

- *Language:* Specify the language for the object. This item has no influence on the object content, but is used for classification purposes and as a search criterion in the object filter. Moreover, certain metadata, such as the date, are displayed in the selected language if they are integrated into the page generated for the object by means of placeholders, so-called WCM tags.
- *Expiration date:* Specify the expiration date (with time) for the object. The expiration date defines how long the object remains valid. The expiration date of the parent topic is used as the default setting.

  Click the arrow icon to open a calendar for selecting the desired date. The time can be set manually. When the object reaches the expiration date in the Edit view, a message is automatically sent to the address specified in the *E-mail Edit* field.
- *Delayed release:* Specify the publication date (with time) for the object. This determines when the object – after having been released by Quality Assurance – will be available for the end users in the Production view. Click the arrow icon to open a calendar for selecting the date. The time can be set manually.
- *Suggested file name:* Enter a name for the page which is generated from the WCM object. In many cases, the WCM system considers this suggestion.
- *Direct release:* You can only select this check box if the functional area “Direct release” has been assigned to you. If you select this check box, the object does not have to be submitted to Quality Assurance; it can be released directly by the responsible editor. The object must only be submitted to Quality Assurance for initial release.

Do one of the following:

- To confirm your entries and make further settings for the new object, click the *Next* button.
- Your entries might already be sufficient for adding the object. In this case, you can click the *OK* button to add the object.

The other steps correspond to the procedure for adding objects by means of the function “New from object type”. See

- “Defining Object Responsibilities” on page 140
- “Entering Attribute Values” on page 142
- “Entering Properties Values” on page 144
- “Selecting a Workflow Transition” on page 145
- “Entering a Remark” on page 146

Importing Linked Objects

You can integrate an existing structure that is linked via references in the WCM system. In this way, you can, for example, import entire tree structures that are tied to each other by links.

Requirements for importing linked objects

For importing linked objects, the following requirements must be met:

- The links must be relative. If the references are absolute, the linked structure cannot be represented in the WCM system.
- The start file of the linked structure must be located at the highest level in the hierarchy. The start file is the file that the WCM system starts from when parsing the linked files.

Required rights and functional areas

You need the following rights and functional areas to add objects by means of the wizard:

- the access rights “Read” and “Create” for the topic to which the object is to be added
- the functional area “Import” and the functional area linked with the object type of the new object. To check your functional areas, choose *Tools → Login info*.

The following sections describe how to proceed when importing linked objects by means of the wizard.

Starting the Wizard

To start the wizard:

1. Change to the Edit view of the website.
2. In the navigation area, mark the topic in which you want to integrate the objects.
3. Choose *Object → Import → linked objects*.

   The wizard starts.
4. Follow the instructions of the wizard that guides you through the import process.

Specifying Title and Object Type; Selecting a File

In this dialog box, you specify the title and object type of the new object. Moreover, you select the file that is the basis for the new objects.

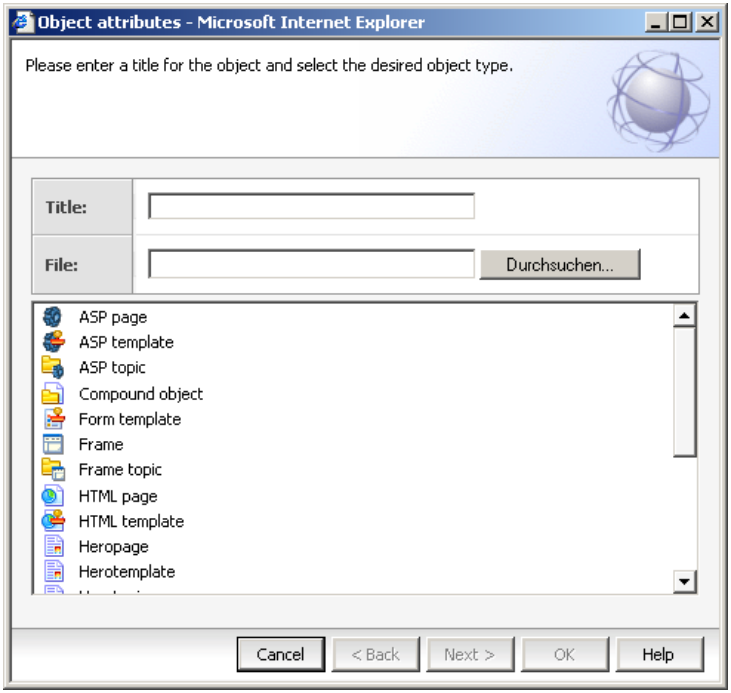


Fig. 55 – Specifying title, file, and object type when importing linked objects

- *Title:* Enter a name for the object. The title is used to identify the object and is displayed both in the header and in the navigation area.

  Entering a title is mandatory.

> **Note:** Titles must not be more than 255 characters long. Also, you should adapt the length of the title in such a way that the resulting file name meets the requirements of your operating system. Under Windows, the absolute path to a topic and a generated page may not be longer than 240 and 260 characters respectively.

- *File:* Click the *Browse* button to select the start file.

> **Note:** Please choose only files that have a file extension.

- Object type: Select the object type from the list, e.g. “HTML page”, “HTML template”, “Topic”Only those object types that match the extension of the specified file and for which an import is possible are displayed.

  Specifying an object type is mandatory. For some object types, it is possible to change the type subsequently.

Do one of the following:

- To confirm your entries and make further settings for the new object, click the *Next* button.
- Your entries might already be sufficient for adding the object. In this case, you can click the *OK* button to add the object.

The other steps correspond to the procedure for adding objects by means of the function “New from object type”. See

- “Specifying Basic Properties” on page 137
- “Defining Object Responsibilities” on page 140
- “Entering a Remark” on page 146

Importing Objects from a ZIP File

You can integrate objects from a ZIP file in the WCM system.

Notes

- In the WCM system, each directory in the ZIP file becomes an object of type “Topic”. This must be considered when creating the ZIP file.
- During the import, content can be added to objects of type “Topic”. For this purpose, save the content as a file in the directory that the WCM system uses for creating the topic. The name of this file must **exactly** correspond to the name specified in the settings of the object type under *Index file*. Ask your WCM administrator for the exact name.
- If the ZIP file contains files with special characters (e.g. German umlauts) in the file name, problems may occur during the import. In this case, use a JAR file instead of a ZIP file. You can use the Jar tool supplied with the Java 2 SDK to create this file.

Required rights and functional areas

You need the following rights and functional areas to add objects by means of the wizard:

- the access rights “Read” and “Create” for the topic to which the object is to be added
- the functional area “Import” and the functional area linked with the object type of the new object. To check your functional areas, choose *Tools* → *Login info*.

The following sections describe how to proceed when importing objects from a ZIP file by means of the wizard.

Starting the Wizard

To start the wizard:

1. Change to the Edit view of the website.
2. In the navigation area, mark the topic in which you want to integrate the objects.
3. Choose *Object* → *Import* → *from ZIP file*.

   The wizard starts.
4. Follow the instructions of the wizard that guides you through the import process.

Selecting a File

In this dialog box, you specify the file that is the basis for adding the objects.

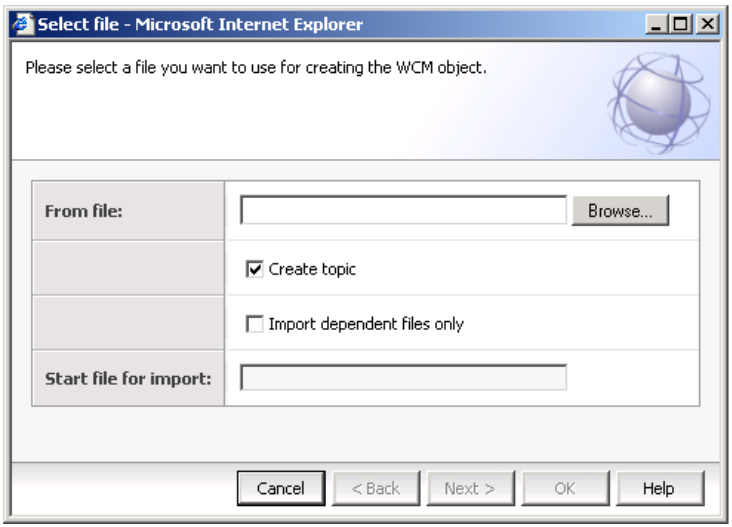


Fig. 56 – Specifying a ZIP file for import

1. Compress the files that you want to import in a ZIP or JAR file (file extension **.zip** or **.jar**).
2. Click the *Browse* button next to the *From file* field to select the ZIP file.
3. Select the desired check boxes:
   - *Create topic:* Select this check box to add a new topic with the name of the ZIP file. If the check box is not selected, the new WCM objects will be added directly below the current topic.
   - *Import dependent files only:* If you select this check box, you must enter the name of the start file or start directory in the field *Start file for import*.

     If you enter the name of a directory, the entire structure below this directory will be imported. Otherwise, only the files that are recursively linked with the start file are imported.
4. Do one of the following:
   - To confirm your entries and make further settings for the new object, click the *Next* button.
   - Your entries might already be sufficient for adding the object. In this case, you can click the *OK* button to add the object.

Specifying Title and Object Type

In this dialog box, you specify the title and object type of the new object.

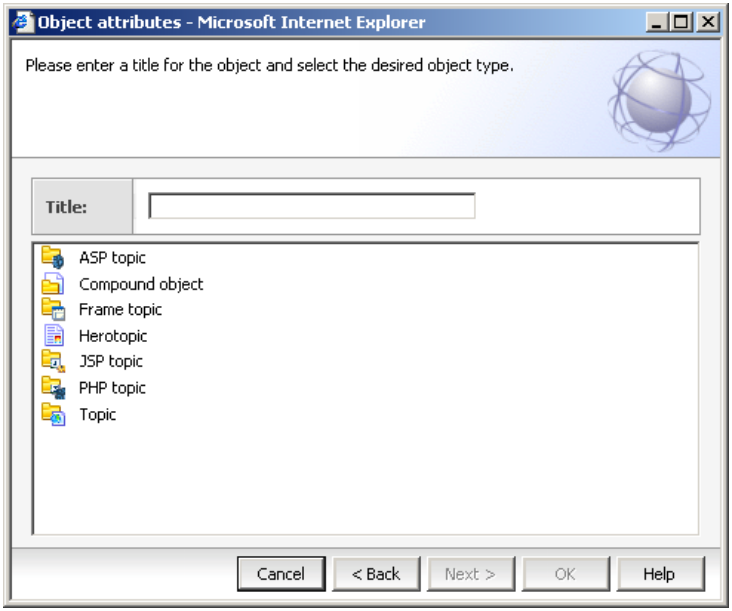


Fig. 57 – Specifying title and type when importing from a ZIP file

- *Title:* Enter a name for the object. The title is used to identify the object and is displayed both in the header and in the navigation area.

  Entering a title is mandatory.

  Note: Titles must not be more than 255 characters long. Also, you should adapt the length of the title in such a way that the resulting file name meets the requirements of your operating system. Under Windows, the absolute path to a topic and a generated page may not be longer than 240 and 260 characters respectively.

- Object type: Select the object's type from the list, e.g. “Frame topic”, “Topic”, “Compound object”The available object types depend on the entries you made in the previous dialog box.

> **Note:** If the start file for the import is a frame object and you select the object type “Topic”, the object type will automatically be changed to “Frame topic” during the import.

Do one of the following:

- To confirm your entries and make further settings for the new object, click the *Next* button.
- Your entries might already be sufficient for adding the object. In this case, you can click the *OK* button to add the object.

The other steps correspond to the procedure for adding objects by means of the function “New from object type”. See

- “Specifying Basic Properties” on page 137
- “Defining Object Responsibilities” on page 140
- “Entering a Remark” on page 146

Copying an Object

Objective

You want to add a copy of the current object at a different location in the object tree.

Prerequisites

- The object has the status “changed”, “rejected”, “released”, or “delayed release”.
- You have the access rights “Read”, “Move and copy”, and “Change metadata” for the object.
- You have the access rights “Read” and “Create” for the topic in which you want to insert the copied object.
- You are assigned the functional area which is linked with the object type of the object. To check your functional areas, choose *Tools* → *Login info*.

Procedure

1. Change to the Edit view of the website.
2. In the navigation area, mark the object that you want to copy (source object).
3. Choose *Edit* → *Copy*.

   The object is copied to the clipboard.
4. In the navigation area, mark the topic in which you want to insert the copy (target topic).
5. Choose *Edit* → *Paste*.

   If the Content client is configured accordingly, a Remark dialog box opens.
6. Enter a remark for the log and click the *OK* button.

   The copy is added to the target topic. Not only the content, but also the metadata and references of the source object are copied. Only the following metadata items are reset:

- “Version”: The main version number of the copy is 0 because the object has been newly created.
- “Author”: When you copy an object, you are the author of the copy.
- “Status”: The status of the copy is “changed” (as for all new objects).

Copying a topic

Note that when you copy a topic, the child objects of that topic are copied, too. However, only those child objects for which you have at least the access right “Read” are copied.

The copies of the topic and the subordinate objects have the major version number 0, the status “changed”, and you are their author.

CHAPTER 6
Editing Objects

In this chapter, the object editing functions are introduced and explained. Object editing includes:

- editing the object content, see section “Editing Objects” on page 177
- maintaining object information, see section “Editing Metadata” on page 193
- maintaining the links between the objects, see section “Maintaining References” on page 201
- setting the access rights, see section “Editing Access Rights” on page 215
- deleting objects, see section “Deleting Objects” on page 227

First, the two basic work steps “Check out” and “Check in” are described (see the following section). At the end of this chapter, you will find information on converting objects (see section “Converting Objects” starting on page 233).

Checking Objects Out and In

In order to edit an object's content and references in the Content client, you have to know the functions “Check out” and “Check in”. This section introduces the following procedures:

- checking out an object with or without the Download applet or the integrated HTML editor, see the following section
- undoing check out of an object whose status is “checked out”, see section “Undoing Check-Out” starting on page 173
- checking in an object with or without the Download applet or the integrated HTML editor, see section “Checking In an Object” on page 174

Checking Out an Object

Before you can edit the content of an object, you must check out the object. Checking out makes the object content available to you for editing and at the same time locks the object for write access by other users. The process of checking out depends on how you configured the Content client.

Prerequisites

- The object has the status “changed”, “rejected”, “released”, or “delayed release”.
- You have the access rights “Read” and “Change object” for the object.
- You are assigned the functional area which is linked with the object type of the object. To check your functional areas, choose *Tools* → *Login info*.

Checking out with Download applet

1. Change to the Edit view of the website.
2. In the navigation area, mark the object that you want to check out.
3. Choose *Edit* → *Check out* or click the appropriate icon.

   Icon for checking out an object

   The file will automatically be saved in your local working directory (the default setting is **c:\temp**; however, you can change this setting in the Configuration dialog box).

   If you have assigned an application to the file type, the checked-out object is opened in the preset editor and can be edited there.

   If no application has been specified for the file type yet, the *Define application* dialog box opens.

   1. Assign the desired application to the file type.
   2. Click the *OK* button.

      The object is transferred to the configured editor and can be edited.

Working with the Download applet for the first time

A security warning is displayed if you work with the Download applet for the first time. You are asked to accept the certificate of Gauss Interprise AG. The security warning for Internet Explorer is shown here as an example.

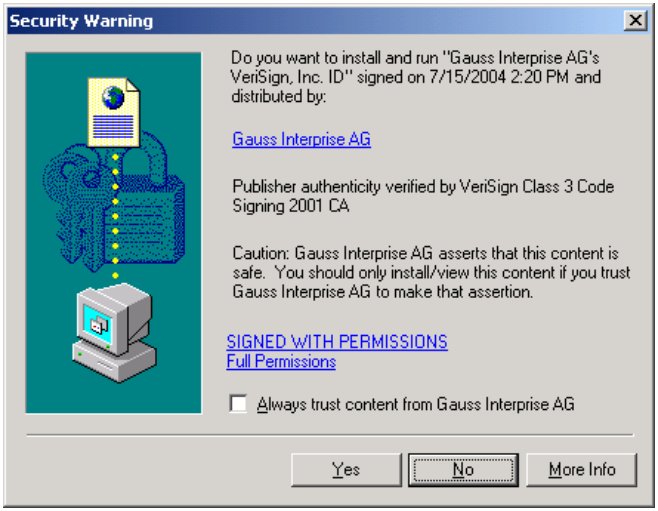

Security Warning
Do you want to install and run "Gauss Interprise AG's VeriSign, Inc. ID" signed on 7/15/2004 2:20 PM and distributed by:
Gauss Interprise AG
Publisher authenticity verified by VeriSign Class 3 Code Signing 2001 CA
Caution: Gauss Interprise AG asserts that this content is safe. You should only install/view this content if you trust Gauss Interprise AG to make that assertion.
SIGNED WITH PERMISSIONS
Full Permissions
Always trust content from Gauss Interprise AG
Yes
No
More Info

Fig. 58 – Security warning when working with the Download applet for the first time

You have the following options:

- *Yes:* Click this button to confirm the security prompt. This is necessary for working with the Download applet.

  If you also select the *Always trust content from Gauss Interprise AG* check box, you accept the certificate. Afterwards, this security prompt will no longer appear when you are working with the Download applet.

- *No:* If you do not confirm the prompt, you cannot work with the Download applet.

- *More Info:* Click this button to request more information about the certificate.

Checking out without Download applet

1. Change to the Edit view of the website.
2. In the navigation area, mark the object that you want to check out.
3. Choose *Edit* → *Check out* or click the appropriate icon.

   Icon for checking out an object

   A window opens that contains a link to the respective file in the export directory of the WCM system.

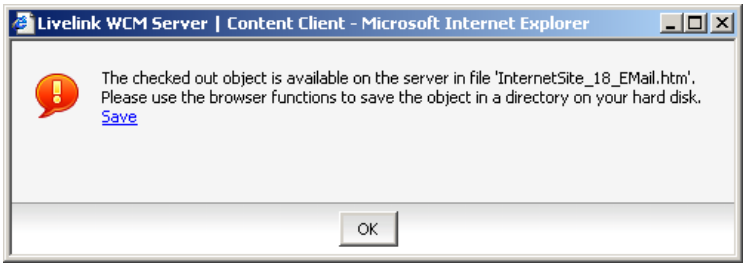

Livelink WCM Server | Content Client - Microsoft Internet Explorer
The checked out object is available on the server in file 'InternetSite_18_EMail.htm'.
Please use the browser functions to save the object in a directory on your hard disk.
Save
OK

Fig. 59 – Saving a file manually

4. Use the browser functions to save the specified file in a directory of your choice (in Internet Explorer, click the right mouse button → *Save Target As*).
5. Close the dialog box by clicking the *OK* button.
6. Open the file from the location where you saved it and edit the file with your preferred editor.

Checking out with integrated HTML editor

1. Change to the Edit view of the website.
2. In the navigation area, mark the object that you want to check out.
3. Choose *Edit* → *Check out* or click the appropriate icon.

   Icon for checking out an object

   The integrated HTML editor starts in a separate window. The content of the checked-out object is transferred to the applet.

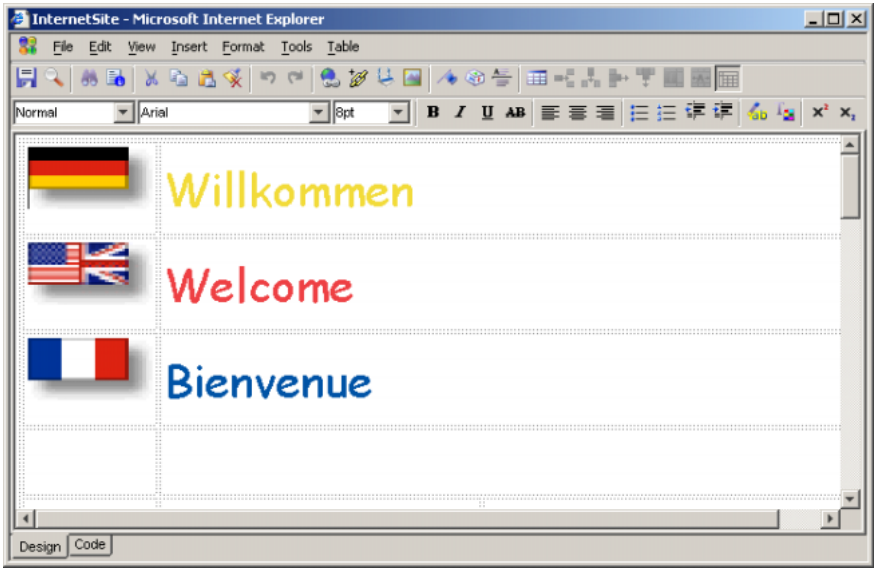


Fig. 60 – The integrated HTML editor

4. Edit the object directly in this window.

   You can use the modes *Design* (object preview similar to browser display) and *Code* (editing the HTML source code) for editing the object.

Undoing Check-Out

Objective

An object was checked out by another user and has not yet been changed. The object's status is “checked out”. You want to edit the object yourself and thus undo the “Check out” action.

Prerequisites

- The object has the status “checked out”.
- You have the access rights “Read” and “Change object” for the object.

Procedure

1. Change to the Edit view of the website.
2. In the navigation area, mark the checked-out object.
3. Choose *Edit* → *Undo* or click the appropriate icon.

   Icon for undoing the action “Check out”

4. Confirm the security prompts that follow.

   This automatically restores the original state of the object.
5. To check out the object yourself, choose *Edit* → *Check out*.

Checking In an Object

After editing the object content, check in the object. The object then has the status “changed” and can be submitted to Quality Assurance. The process of checking in depends on how you configured the Content client.

> **Note:** We recommend that you not change the deployment system between checking an object out and in. Changing the deployment system causes problems with objects that contain references to other objects.

Prerequisites

- The object has the status “checked out”.
- You have the access rights “Read” and “Change object” for the object.
- You are assigned the functional area which is linked with the object type of the object. To check your functional areas, choose *Tools* → *Login info*.

Checking in with Download applet

1. Save your changes and close the editor.
2. In the navigation area, mark the object that you edited.
3. Choose *Edit* → *Check in* or click the appropriate icon.

   Icon for checking in an object

   When the object is checked in, the file is automatically read from your local working directory and transferred to the server.

   At the same time, the system checks whether the file checked in contains embedded objects (e.g. images). If there are embedded objects, an according dialog box is displayed.

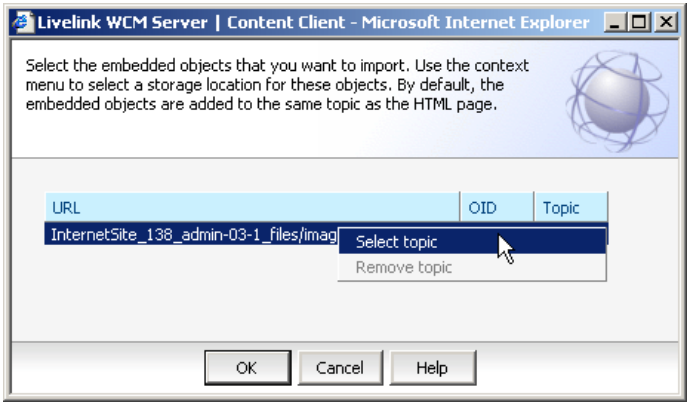

Livelink WCM Server | Content Client - Microsoft Internet Explorer
Select the embedded objects that you want to import. Use the context menu to select a storage location for these objects. By default, the embedded objects are added to the same topic as the HTML page.
URL
OID
Topic
InternetSite_138_admin-03-1_files/imag
Select topic
Remove topic
OK
Cancel
Help

Fig. 61 – Selecting embedded objects when checking in an object

In this dialog box, you determine which embedded objects are to be imported into the WCM system together with the checked-in file. You can deselect individual – or all – objects. These objects will not be imported. If an object with the same name and object type already exists in your website, the *OID* column shows the OID of this object. Deselect existing objects to prevent adding multiple objects containing the same file, e.g. the same image.

By default, the embedded objects are added to the topic containing the referencing object. To specify a different storage location for individual objects, choose *Select topic* on the context menu and mark the desired topic in the selection dialog box. The *Topic* column will show the OID of the storage location.

4. Select the desired objects and click the *OK* button.
5. Enter a remark for the log and click the *OK* button.

Checking in without Download applet

1. Save your changes and close the editor.
2. In the navigation area, mark the object that you edited.
3. Choose *Edit* → *Check in* or click the appropriate icon.

   Icon for checking in an object

4. The *Select file* dialog box opens.
5. Click the *Browse* button and navigate to the edited file.

   If instead of the file originally saved, you accidentally (or deliberately) select a different file, this is equivalent to replacing the object. Livelink WCM Server is not able to verify whether the file is identical to the file you previously saved.
6. To enter a remark for the log, click the *Next* button.
7. Enter the remark in the Log dialog box.
8. Click the *OK* button.

Checking in with integrated HTML editor

1. Choose *File* → *Save* in the integrated HTML editor or click the appropriate icon.

   Icon for saving

2. Close the integrated HTML editor.

3. Choose *Edit* → *Check in* or click the appropriate icon.

   Icon for checking in an object

   During the check-in, the changed content is transferred to the server.

4. Enter a remark for the log and click the *OK* button.

Editing Objects

In this section, you learn which functions are required for editing object content. The following procedures are introduced:

- editing the object content of an existing object, see the following section
- editing an object you already checked out, see section “Editing Object Content Again” starting on page 184
- undoing changes made to an object, see section “Undoing Changes” starting on page 186
- moving an object to a different location in the topic structure of the website, see section “Moving an Object” starting on page 187
- updating an object that has reached its expiration date, see section “Updating an Expired Object” starting on page 189
- replacing an object with an external file, see section “Replacing the Content of an Object” starting on page 190

Editing Object Content

Objective

You want to edit an existing object in your website.

Prerequisites

- The object has the status “changed”, “rejected”, “released”, or “delayed release”.
- You have the access rights “Read” and “Change object” for the object.

Procedure

1. Change to the Edit view of the website.
2. In the navigation area, mark the object that you want to edit and check the object out.

   Depending on the configuration of the Content client, the object is opened in the integrated HTML editor or in the associated editor program.
3. Edit the object in the respective editor.
4. When you have finished editing the object, check the object back in to the WCM system.

   The object has the status “changed”.
5. To submit the object to Quality Assurance, choose *Object* → *Submit* or click the appropriate icon.

   Icon for submitting an object

Inserting a WCM Tag in an HTML Object

For editing HTML objects, the Content client provides the integrated HTML editor. In addition to the familiar HTML editing features, the integrated HTML editor provides an easy-to-use function for conveniently inserting WCM tags in HTML objects. A wizard guides you through this process. WCM tags are placeholders enabling you to integrate information from the object's metadata, such as the expiration date, into the page generated for the object.

Starting the wizard

To start the wizard for inserting WCM tags:

1. Make sure that the *Use integrated HTML editor* check box in the Configuration dialog box is selected.
2. In the navigation area, mark the HTML object that you want to edit and check the object out.

   The object's content is transferred to the integrated HTML editor.
3. Choose *Insert* → *Insert WCM tag* or click the appropriate icon.

   Icon for inserting a WCM tag

Selecting a metadata item

In the first dialog box, you specify which object information you want to integrate in the page by means of the WCM tag.

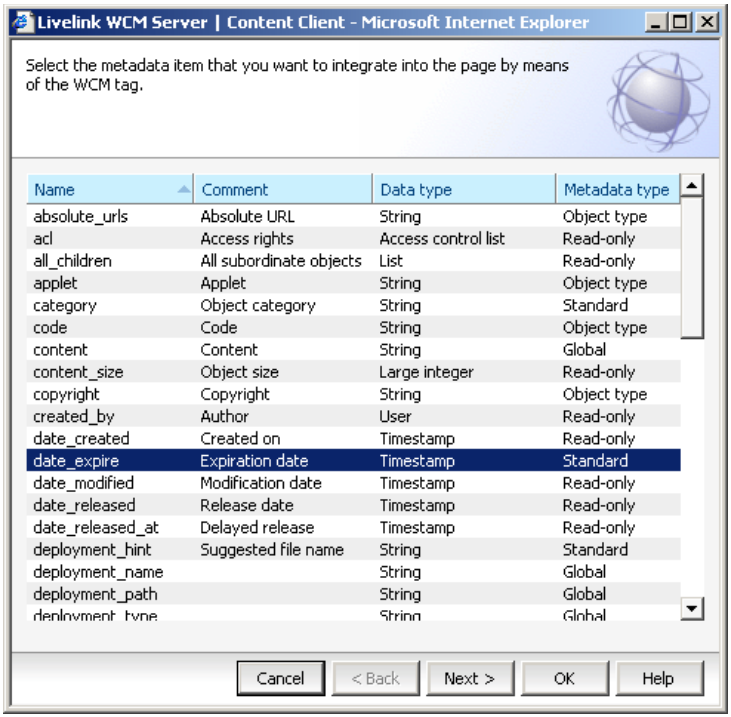

Livelink WCM Server | Content Client - Microsoft Internet Explorer
Select the metadata item that you want to integrate into the page by means of the WCM tag.
Name
Comment
Data type
Metadata type
absolute_urls Absolute URL String Object type
acl Access rights Access control list Read-only
all_children All subordinate objects List Read-only
applet Applet String Object type
category Object category String Standard
code Code String Object type
content Content String Global
content_size Object size Large integer Read-only
copyright Copyright String Object type
created_by Author User Read-only
date_created Created on Timestamp Read-only
date_expire Expiration date Timestamp Standard
date_modified Modification date Timestamp Read-only
date_released Release date Timestamp Read-only
date_released_at Delayed release Timestamp Read-only
deployment_hint Suggested file name String Standard
deployment_name String Global
deployment_path String Global
Cancel
< Back
Next >
OK
Help

Fig. 62 – Selecting a WCM tag

The dialog box shows the metadata of the current object. For each metadata item, the list provides the corresponding WCM tag, the name in the Content client, the data type, and the metadata type. Metadata types include:

- “Standard” = standard metadata item
- “Category” = property from the assigned object category
- “Object type” = attribute from the attribute set assigned to the object type

- “Global” = global, not object-specific information (e.g. website the object belongs to)
- “Read-only” = the metadata item cannot be changed

Do one of the following:

- To insert the WCM tag, click the *OK* button.
- WCM tags can be equipped with parameters (exception: metadata of type “Global”). To specify parameters, click the *Next* button.

> **Note:** For detailed information on parameters in WCM tags, refer to section “Parameterized WCM Tags” starting on page 440.

Specifying parameters for the WCM tag

In this dialog box, you can specify parameters for the WCM tag. Parameters in WCM tags fulfill multiple tasks. You can use them to format the integrated object information or to activate special functions.

The type and number of available parameters depends on the metadata item's data type. The following figure shows the parameters for the metadata item “Expiration date” which has the data type “Timestamp”.

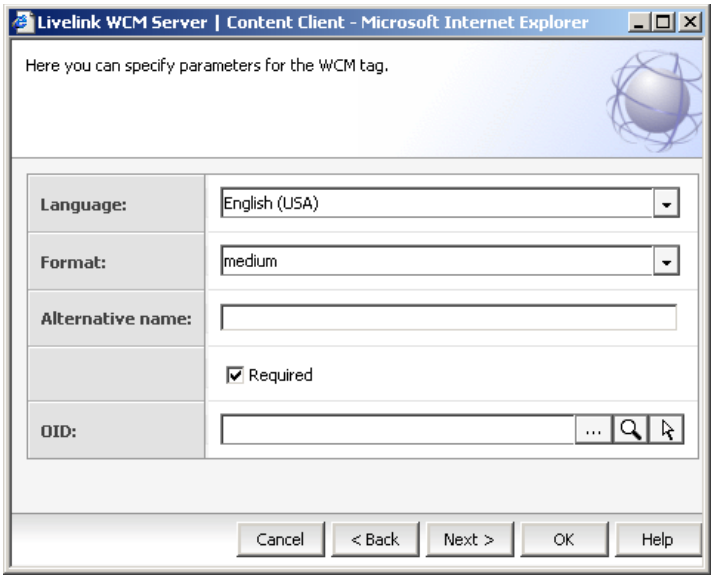


Fig. 63 – Specifying parameters for a WCM tag

The following sections provide detailed information on the individual parameters:

- on *Language*, see section “The `locale` Parameter” on page 449
- on *Format*, see section “The `format` Parameter” on page 441
- on *Alternative name*, see section “The `label` Parameter” on page 453
- on *Required*, see section “The `required` Parameter” on page 453
- on *Tag type*, see section “The `type` Parameter” on page 450
- on *OID*, see section “Referencing Objects” on page 457

Specify the desired values for the parameters and do one of the following:

- To insert the WCM tag, click the *OK* button.
- To add more parameters to the WCM tag, click the *Next* button.

Specifying the InSite Editing parameters

In this dialog box, you specify if and how the integrated metadata item can be edited with InSite Editing.

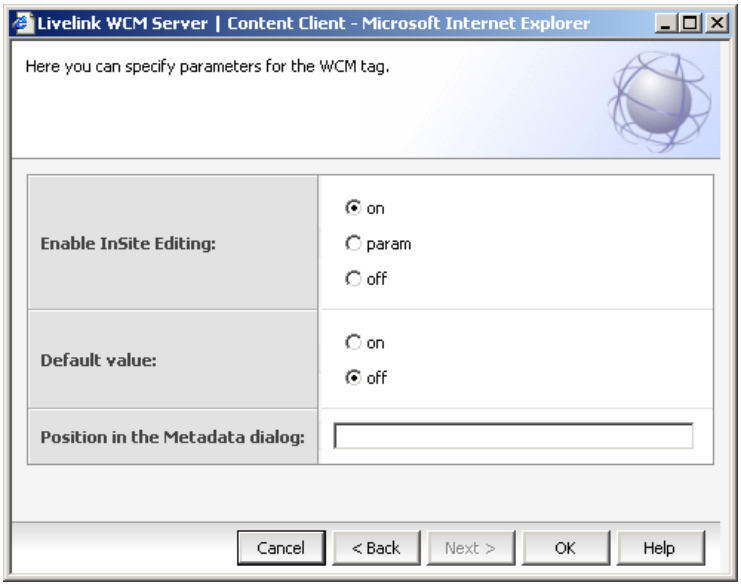


Fig. 64 – Specifying additional parameters for a WCM tag

- *Enable InSite Editing:* Use these radio buttons to control InSite Editing for the integrated metadata item. Enabling InSite Editing does not make sense for metadata items of type “Read-only”. Possible values for the parameter are:

- *on:* The metadata item is equipped with a toolbar and highlighted by a red frame. You can edit the information directly in the website or in the WCM tag dialog box of InSite Editing.
- *param:* You can edit the metadata item in the WCM tag dialog box of InSite Editing. It will not be highlighted in the website.
- *off:* InSite Editing is not enabled for this metadata item (default setting).

- *Default value* (for template objects only): By means of this parameter, you can specify a default value for the integrated metadata item. Users who edit the objects based on this template can use this default value as a hint regarding the information they are expected to enter in this field.
  - *on:* Users are able to specify a default value for this metadata item in the Metadata dialog box.
  - *off:* Users cannot specify a default value for this metadata item.
- *Position in the Metadata dialog:* Use this parameter to specify the position of the WCM tag in the WCM tag dialog box of InSite Editing. This enables you to position the input fields for essential information on top of the dialog box. You can only enter integers.

To insert the WCM tag, click the *OK* button.

Editing Object Content Again

Objective

You want to open an object you have already checked out in the editor in order to make further changes.

Prerequisites

- The object has the status “checked out”.
- You checked out the object yourself.

Procedure

1. Change to the Edit view of the website.
2. In the navigation area, mark the checked-out object.
3. The next step depends on how you configured the Content client.
   - If you have selected the *Use Download applet* check box in the Configuration dialog box, choose *Edit* → *Edit content* or click the appropriate icon.

     Icon for editing the content of an object

     The object will open in the application you assigned.
   - If you have cleared the *Use Download applet* check box, simply open the file again from its local storage location, make the changes and save the file.
4. When you have finished editing the object, check the object back in to the WCM system.

   The object has the status “changed”.
5. To submit the object to Quality Assurance, choose *Object* → *Submit* or click the appropriate icon.

   Icon for submitting an object

Undoing Changes

Objective

You want to undo changes made to an object, i.e. you want to restore an earlier version of the object.

Prerequisites

- The object has the status “changed”, “rejected”, “released”, or “delayed release”.
- You have the access rights “Read”, “Change object”, and “Change metadata” for the object.
- You are assigned the functional area “Log dialog”. To check your functional areas, choose *Tools* → *Login info*.

Procedure

1. Change to the Edit view of the website.
2. In the navigation area, mark the object for which you want to restore an earlier version.
3. Choose *View* → *Log*.

   The log dialog box opens. This dialog box lists all available object versions.
4. In the Log dialog box, select the version that you want to restore.
5. By means of the right mouse button → *Display version*, check whether you have selected the right version.
6. Choose right mouse button → *Restore old version*.
7. Confirm the security prompt by clicking the *Yes* button.

   The entry “Earlier version restored” is written to the log. The number of the restored version is entered in the *Remark* column.

After the earlier version of the object has been restored, the object has the status “changed”.

8. To submit the object to Quality Assurance again, choose *Object* → *Submit* or click the appropriate icon.

Icon for submitting an object

Notes:

In the Admin client, the WCM administrator can determine the maximum number of restorable versions. For detailed information, refer to the Livelink WCM Server Administrator Manual.

The following standard metadata of an object version are restored: Content, Type, Title, Heading, Suggested file name, Direct release, Language, Keywords, Description, Target group, E-mail Edit, E-mail QA, E-mail Release.

Moving an Object

Although easy to do, moving objects is a complex process. It does not merely move a file from one directory to another. It also adapts all references to the object. If the object is a topic, all references to the child objects are also adapted.

Objective

You want to move an object to a different location in the website’s topic structure.

Prerequisites

- The object has the status “changed”, “rejected”, “released”, or “delayed release”.

When a topic is moved, its subordinate objects are also moved. For this reason, the subordinate objects must also have the status “changed”, “rejected”, “released”, or “delayed release”.

- You have the access rights “Read”, “Move and copy”, and “Change metadata” for the object.
- You have the access rights “Read” and “Create” for the topic in which you want to insert the object.

Procedure

1. Change to the Edit view of the website.
2. In the navigation area, mark the object that you want to move (source object).
3. Choose *Edit* → *Cut*.
4. In the navigation area, mark the topic to which you want to move the object (target topic).
5. Choose *Edit* → *Paste*.

   If the Content client is configured accordingly, a Remark dialog box opens.
6. Enter a remark for the log and click the *OK* button.

   The object is moved to the target topic. Not only the content, but also the metadata and references of the source object are copied. The references are updated to reflect the new location.

   The status of the moved object is “changed”. After a topic has been moved, it also has the status “changed”. The action does not change the status of the subordinate objects.
7. To submit the moved object to Quality Assurance again, choose *Object* → *Submit* or click the appropriate icon.

Icon for submitting an object

Updating an Expired Object

Objective

The WCM system has notified you that an object has passed its expiration date. You now want to update the object.

Prerequisites

- The object has the status “changed”, “rejected”, “released”, or “delayed release”.
- You have the access rights “Read” and “Change metadata” for the object.

Procedure

1. Change to the Edit view of the website.
2. In the navigation area, mark the object that you want to update.

   You can also use the object filter to find expired objects.
3. Update the object content (see section “Editing Object Content” on page 178).

   Once you have updated the content, you need to change the expiration date. This is done in the Metadata dialog box.
4. Choose *View* → *Metadata*.
5. In the *Expiration date* field on the *General* tab, specify how long the updated object is to remain valid. Click the arrow icon to open a calendar for selecting the desired date. The time can be set manually.
6. Click the *OK* button.

7. To submit the updated object with the new release date to Quality Assurance, choose *Object* → *Submit* or click the appropriate icon.

   Icon for submitting an object

Replacing the Content of an Object

Objective

You want to replace an object that already exists in the WCM system with an external file that is stored on your local hard disk or in your network.

Prerequisites

- The object has the status “changed”, “rejected”, “released”, or “delayed release”.
- You have the access rights “Read” and “Change object” for the object.

Procedure

1. Change to the Edit view of the website.
2. In the navigation area, mark the object that you want to replace.
3. Choose *Edit* → *Replace*.

   The *Select file* dialog box opens.

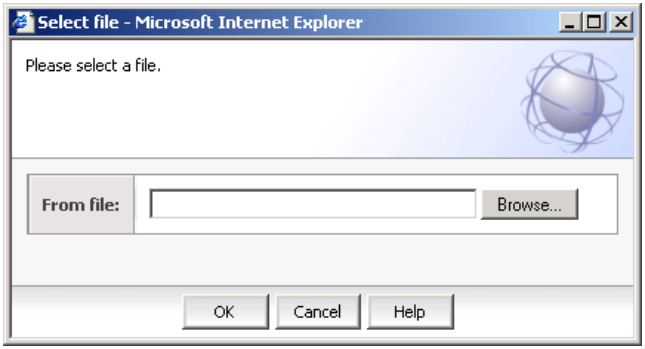


Fig. 65 – Dialog box for replacing an object

4. Click the *Browse* button.

   The dialog box for selecting files opens.

5. Navigate to the file that is to replace the object and click the *Open* button.

   > **Note:** Make sure that the file you selected has a file type that matches the object type of the object to be replaced. For example, if you want to replace an object of type “GIF image”, you must select a file of the file type “GIF”. Otherwise, an error message appears. This error can only be remedied by selecting *Edit* → *Undo*.

   The file you selected appears in the *Select file* dialog box.

6. Click the *OK* button.

   If you are working **with the Download applet**, the system checks whether the selected file contains embedded objects. If there are embedded objects, an according dialog box is displayed.

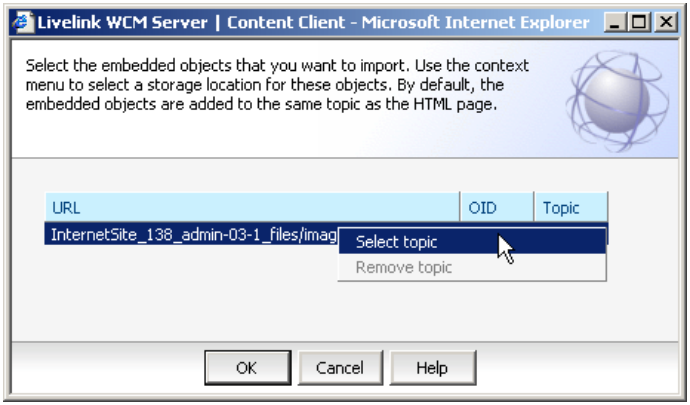

Livelink WCM Server | Content Client - Microsoft Internet Explorer
Select the embedded objects that you want to import. Use the context menu to select a storage location for these objects. By default, the embedded objects are added to the same topic as the HTML page.
URL
OID
Topic
InternetSite_138_admin-03-1_files/imag
Select topic
Remove topic
OK
Cancel
Help

Fig. 66 – Selecting embedded objects when replacing an object

In this dialog box, you determine which embedded objects are to be imported into the WCM system together with the selected file. You can deselect individual – or all – objects. These objects will not be imported. If an object with the same name and object type already exists in your website, the *OID* column shows the OID of this object. Deselect existing objects to prevent adding multiple objects containing the same file, e.g. the same image.

By default, the embedded objects are added to the topic containing the referencing object. To specify a different storage location for individual objects, choose *Select topic* on the context menu and mark the desired topic in the selection dialog box. The *Topic* column will show the OID of the storage location.

7. Select the desired objects and click the *OK* button.
8. Enter a remark for the log and click the *OK* button.
9. To submit the object to Quality Assurance, choose *Object* → *Submit* or click the appropriate icon.

Icon for submitting an object

Editing Metadata

For general information on the metadata, see section “Metadata” starting on page 42.

In this section, the following procedures are introduced:

- editing the metadata of one object, see the following section
- simultaneously editing the metadata of several objects, see section “Editing Metadata of Several Objects” starting on page 196
- time-controlled publication of an object, see section “Setting the Delayed Release Option for an Object” starting on page 198
- changing the type of an object, see section “Changing the Object Type” starting on page 199
- integrating metadata in a page, see section “Integrating Metadata in a Page” on page 200

Editing an Object’s Metadata

Objective

You want to edit an object’s metadata. For editing the metadata of an individual object, the Metadata dialog box is available.

Prerequisites

- The object has the status “changed”, “rejected”, “released”, or “delayed release”.
- You have the access rights “Read” and “Change metadata” for the object.

Procedure

1. Change to the Edit view of the website.
2. In the navigation area, mark the object whose metadata you want to edit.
3. Choose *View* → *Metadata*.

   In the right window pane, the metadata is displayed on various tabs.

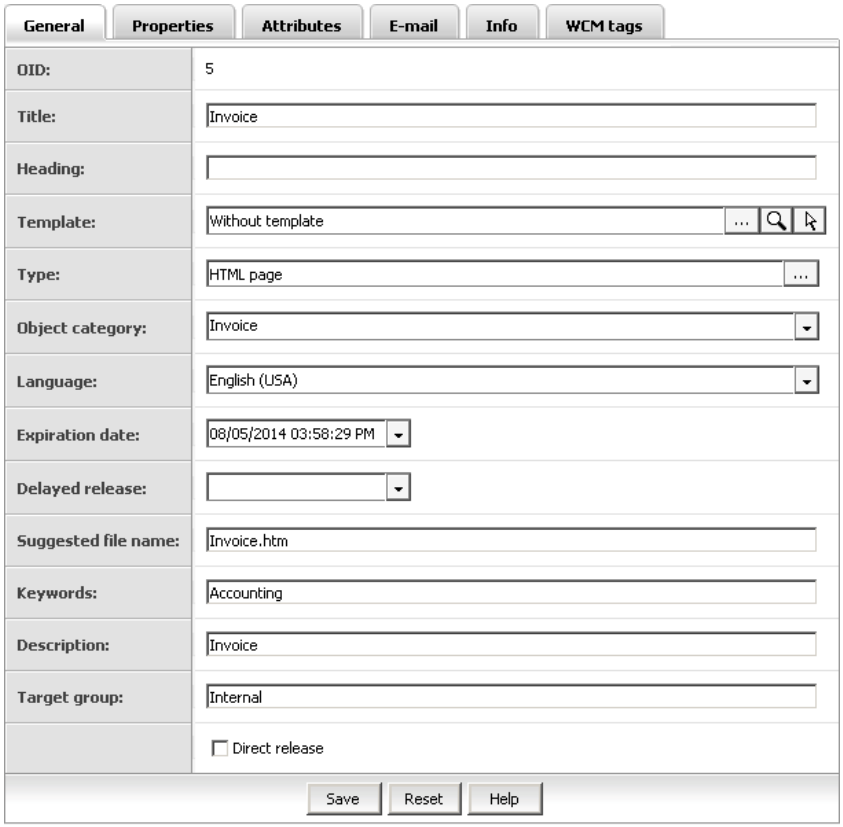


Fig. 67 – The Metadata dialog box

- *General* tab: All entries on this tab are standard metadata that are defined by the WCM system and are available for each WCM object.
- *Properties* tab (only displayed if an object category is assigned to the object): Properties are metadata that do not belong to the standard metadata. The properties of the object result from its assignment to an object category.
- *Attributes* tab (only displayed if the object type is linked with an attribute set): Attributes are metadata that do not belong to the standard metadata. The attributes of the object result from its association with an object type.
- *E-mail* tab: On this tab, the e-mail addresses specified for the object are displayed. You can send messages to these e-mail addresses when the object acquires a certain status.
- *Info* tab: This tab is only used for information purposes. The standard metadata displayed here are controlled by the WCM system.
- *WCM tags* tab: This tab is only used for information purposes. It shows all metadata that are integrated into the content of the current object by means of WCM tags.

4. Make the desired changes on the tabs.

> **Note:** It is not possible to change the object category and the properties of an object in **one** step. Change the object category first and save your changes. Then, you can edit the properties of the object. The same applies to the object type and the attributes of an object.

5. Click the *Save* button.

   The metadata are changed. The object has the status “changed”.

6. To submit the object to Quality Assurance, choose *Object* → *Submit* or click the appropriate icon.

   Icon for submitting an object

Editing Metadata of Several Objects

Objective

You want to change the metadata of several objects simultaneously. A wizard guides you through this process.

Prerequisites

- The objects have the status “changed”, “rejected”, “released”, or “delayed release”.
- You have the access rights “Read” and “Change metadata” for the objects.

Procedure

1. Change to the Edit view of the website.
2. Choose *View* → *Object list* or use the object filter, e.g. if you want to set a new (identical) expiration date for all expired objects.
3. In the object list or hitlist, mark the objects whose metadata you want to edit.
4. Choose *Edit* → *Edit metadata of current objects*.

   The wizard starts. Here you can

   - change basic metadata (e.g. the template, the object type, and the publication date)
   - change the object responsibilities and descriptions

- change the attributes that the objects possess due to their association with an object type

For detailed information on the individual entries, refer to the online help of the Content client.

5. Make the desired changes and save them.

> **Notes:**
>
> If a field is left empty, the individual settings of the objects will remain unchanged.
>
> This dialog box for modifying the attributes is only displayed if you have explicitly selected an object type in the first dialog box of the wizard.

The system checks whether the individual objects have the required status (“changed”, “rejected”, “released”, or “delayed release”). Moreover, the system checks whether you have the required access rights. Objects that do **not** meet both requirements are removed from the selection.

The metadata of all objects that meet both requirements are changed. The objects acquire the status “changed”.

6. To submit the objects to Quality Assurance, choose *Object* → *Submit* or click the appropriate icon.

Icon for submitting an object

> **Note:** If an error occurs while the metadata are changed, the entire action is aborted. The changes performed so far are undone. In this case, you receive an error message.

Setting the Delayed Release Option for an Object

Objective

You want to specify a publication date for an object. You do not want the object to be published in the Production view directly after its release by Quality Assurance, i.e. it is not to be published until the date specified.

Prerequisites

- The object has the status “changed”, “rejected”, “released”, or “delayed release”.
- You have the access rights “Read” and “Change metadata” for the object.

Procedure

1. Change to the Edit view of the website.
2. In the navigation area, mark the object for which you want to set a delayed release date.
3. Choose *View* → *Metadata*.
4. Enter the desired publication date in the *Delayed release* field on the *General* tab.

   Click the arrow icon to open a calendar for selecting the date. The time can be set manually.
5. To submit the object to Quality Assurance, choose *Object* → *Submit* or click the appropriate icon.

   Icon for submitting an object

   After having been released by Quality Assurance, the object has the status “delayed release”, which is marked by the icon . When the publication date is reached, the object is published in the Production view.

Notes on changing and deleting the publication date

- You can change or delete the publication date of an object that has not yet been released by Quality Assurance via the Metadata dialog box.
- You can change or delete the publication date of an object that has the status “delayed release” via the Metadata dialog box. This change must be submitted to Quality Assurance and released. Thus, the original publication date becomes invalid.
- As soon as the publication date of an object is reached, the respective entry in the Metadata dialog box is deleted by the system and the object version is published in the Production view. Changing the publication date would only affect the next object version. The object version already published remains visible.

Changing the Object Type

Objective

You want to change an object's type.

Prerequisites

- The object has the status “changed”, “rejected”, “released“, or “delayed release”.
- You have the access rights “Read” and “Change metadata” for the object.
- You are assigned the functional area that is linked with the desired object type. To check your functional areas, choose *Tools* → *Login info*.

Notes

- The custom object types created by the administrator in the Admin client **cannot** be converted into other object types.
- Changing the object type is only possible for some of the default object types. For an overview of the changes possible from a technical point of view, refer to the *Type* dialog box of the Metadata dialog box.

Procedure

1. Change to the Edit view of the website.
2. In the navigation area, mark the object whose type you want to change.
3. Choose *View* → *Metadata*.
4. In the *Type* field on the *General* tab, select the new object type for the object.
5. Click the *OK* button.
6. To submit the object to Quality Assurance, choose *Object* → *Submit* or click the appropriate icon.

Icon for submitting an object

Integrating Metadata in a Page

You can integrate any of the object's metadata in a document or in a template by means of WCM tags. The WCM tags for the metadata are available in the Metadata dialog box. They are displayed as tooltips when you move the mouse over the names of metadata items.

The following code sample illustrates how an object's expiration date is integrated in a page by means of the appropriate WCM tag. As a result, the HTML page will always display the current expiration date setting.

```
<body>
      ...
      <P>Document valid until: {VIPDATE_EXPIRE}</p>
      ...
</body>
```

For detailed information on using WCM tags, refer to chapter 15 “Working with WCM Tags”.

Maintaining References

For general information on references, refer to section “References” starting on page 46.

Valid references are an important precondition for successfully using a website. For this reason, maintaining references is one of the most important tasks when managing a website.

The references of a WCM object include:

- integrated graphics files
- links to other objects within the website
- references to external addresses outside the current website (e.g. pages on the Internet)

For the most object types, the internal reference check guarantees consistency of the internal references contained in the metadata and the references included in the page. The References dialog box shows all references of an object.

This section describes how to

- work with the References dialog box, see the following section
- add references in the Content client or in an editor, see section “Adding References” starting on page 203
- delete references, see section “Deleting References” starting on page 214

The References Dialog Box

Note: For viewing the References dialog box, you require the functional area “References dialog”. To check your functional areas, choose *Tools* → *Login info*.

Choose *View* → *References* to open the References dialog box.

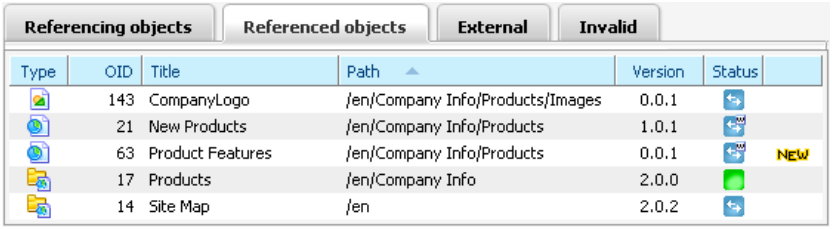


Fig. 68 – The References dialog box

On four tabs, you can view the references of the current object. References realized by means of special attributes are also displayed here:

- *Referencing objects:* objects containing references to the current object (the object is the target of the reference)
- *Referenced objects:* objects referenced by the current object (the object is the source of the reference)

- *External:* references to objects and pages located outside the website
- *Invalid:* references that no longer work (e.g. references to deleted objects)

Note: The Reference dialog box does not display references from an assigned template and references outside the body element that are overwritten by template code.

The *Referencing objects* and *Referenced objects* tabs display the references in an object list. For each object, the object type, OID, title, path to the root object, version, and status are displayed. By clicking on the header row of a column, you can sort the object list according to the criterion in this column. To reverse the sort order, click the header row again. This way you can clearly arrange long lists.

You can navigate to the relevant object by double-clicking on an entry in the object list. If you select an entry on the *External references* tab, the referenced page will open in a separate browser window.

References realized by means of special attributes are marked by the icon . If you move the mouse over the icon, the name of the attribute is displayed as a tooltip.

Adding References

There are different ways of adding references to an object:

- by means of the function “Add reference” in the Content client
- by means of the function “Add reference to a WCM object” in the integrated HTML editor
- directly in the editor

Adding References in the Content Client

Objective

You want to add a reference to a WCM object which points to another object within the same website.

Prerequisites

- The object has the status “changed”, “rejected”, “released”, or “delayed release”.
- You have the access rights “Read” and “Change metadata” for the object.
- You are assigned the functional area “References dialog”. To check your functional areas, choose *Tools* → *Login info*.

Procedure

1. Change to the Edit view of the website.
2. In the navigation area, mark the object to which you want to add a reference.
3. Choose *Edit* → *Add reference*.

   A dialog box with the topic structure of the website opens.

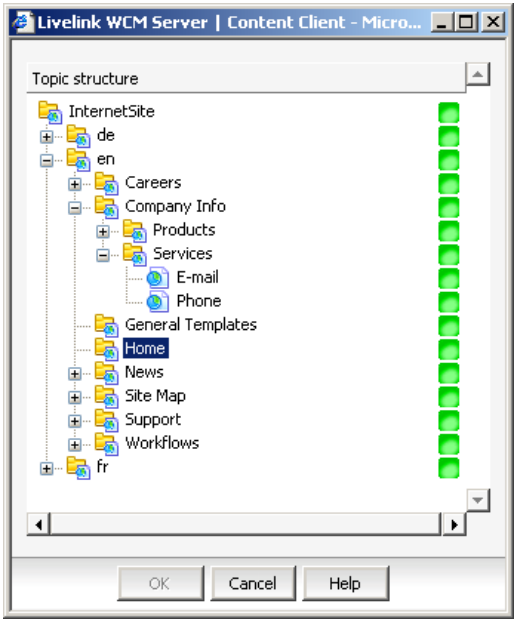


Fig. 69 – Topic structure when adding references

4. In the topic structure, select the object that is to be the target of the reference.

   The reference is registered in the WCM system. It is displayed in the References dialog box on the *Referenced objects* tab and is marked as *NEW*.

5. To add the reference to the object content, check out the object.
6. Use the editor functions to move the automatically inserted link to the desired position and edit the link text, if required.
7. When you have finished editing the object, check the object back in to the WCM system.

References are checked automatically when the object is checked in to the WCM system. The system compares the content of the HTML file with the metadata of the object. The following items are checked:

- whether all references contained in the object's metadata are present on the page (as links or integrated graphics)
- whether all links and integrated graphics present on the page are also contained as references in the object's metadata

The reference check automatically resolves any inconsistencies between the metadata and the object content.

Adding References in the Integrated HTML Editor

Objective

You want to add a reference to a WCM object which points to another object within the same website.

Prerequisites

- The object has the status “changed”, “rejected”, “released”, or “delayed release”.
- The object is an HTML object.
- You have the access rights “Read” and “Change metadata” for the object.
- The *Use integrated HTML editor* check box in the Configuration dialog box is selected.

Procedure

1. Change to the Edit view of the website.
2. In the navigation area, mark the object that you want to edit and check the object out.

   The object's content is transferred to the integrated HTML editor.
3. Choose *Insert* → *Add reference to a WCM object* or click the appropriate icon.

   Icon for adding references

   A dialog box with the topic structure of the website opens.

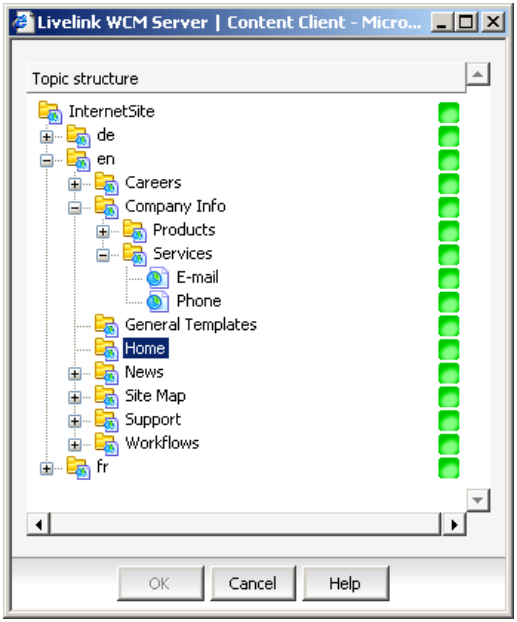


Fig. 70 – Topic structure when adding references

4. In the topic structure, select the object that is to be the target of the reference.
5. Move the inserted link to the desired position and edit the link text, if required.
6. When you have finished editing the object, check the object back in to the WCM system.

   References are checked automatically when the object is checked in to the WCM system. References to WCM objects are entered in the reference management of Livelink WCM Server. The reference check automatically resolves any inconsistencies between the meta-data and the object content.

Inserting an image

Choose *Insert* → *Insert Image* in the integrated HTML editor to integrate an image in an object. If you use WebDAV for editing your objects, you cannot only insert image files stored on your local hard disk or a network drive, but also images from web servers outside the company network. In the *Insert Image* dialog box, click the following icon:

Icon for inserting an image file via WebDAV

Adding References in the Editor

Objective

You want to add a reference to an object which points to another WCM object in the same website or to an external address.

Prerequisites

- The object has the status “changed”, “rejected”, “released”, or “delayed release”.
- You have the access rights “Read”, “Change object”, and “Change metadata” for the object.

Procedure

1. Change to the Edit view of the website.
2. In the navigation area, mark the object to which you want to add a reference and check the object out.
3. Insert a reference at the desired position in the object content.

   Note: Do **not** use the tag `<BASE href>` in the head element of the HTML page. Using this tag interferes with the reference management of Livelink WCM Server.

   To reference an object from the same website, select the target of the reference in the navigation area.

   For conveniently adding references to WCM objects, you can use the following commands from the *Edit* menu or the context menu.

- *Copy URL*
  This command creates a reference to the object you have marked in the navigation area. This reference is saved in the form of the object's URL and can be inserted into the object via the clipboard.
- *Copy HTML tag*
  This command creates a complete HTML reference to the object you have marked in the navigation area. The reference has the format `<A HREF={URL of the object}>{title of the object}</A>` and can be inserted into the object via the clipboard.

> **Note:** The procedure in the Mozilla browser differs from this description. The reference is not copied to the clipboard. A separate window with the URL or the HTML tag opens instead. Use the context menu of the browser to copy the reference to the clipboard.

4. When you have finished editing the object, check the object back in to the WCM system.

   When the object is checked in, the WCM system automatically recognizes references that have been inserted manually and that contain the HTML tags for references (`<a href="...">`, `<img src="...">`, `<frame src="...">`) and adds these references to the metadata.

   As when adding references in the Content client, references are checked automatically when the object is checked in to the WCM system.

Special cases: JSP pages and JavaScript code

In some cases, the WCM system cannot identify references automatically (e.g. in the JSP code fragment `String url = "...";`). To solve this problem, you can precede the reference with the expressions in the following table.

Table 13 – Expressions for identifying references

Expression	Application area
Code fragment	
`/*VIPURL*/`	This expression can be used in the code wherever comments are allowed.

Java
`String url = /*VIPURL*/"{URL of the object}";`
JavaScript
`var url = /*VIPURL*/"{URL of the object}";`	

`getVIPURL`	This expression can be used in the code wherever method calls and function calls are allowed.
JavaScript `var url = getVIPURL("{URL of the object}");`	
`{attribute name}VIPURL`	You have developed a custom JSP tag. If the value of an attribute of this JSP tag is to be interpreted as a reference, the attribute name must contain the suffix “VIPURL”. **Note:** Each value of an attribute whose name ends with “VIPURL” is interpreted as a reference. This has to be considered when naming attributes.
JSP `<taglib:myTag myAttributeVIPURL = "{URL of the object}" />`	

JSP example

The following JSP example illustrates the use of expressions for identifying references.

```
<html>
<head>
<title>{VIPTITLE}</title>

<%!
      static String url = /*VIPURL*/"http://wcmserver.company.example/
         InternetSite_Edit/InternetSite/ProductOverview.html";

%>
</head>
<body>

      <a onclick='goto("<%= url %>")'>Go to 'Product Overview'
         </a><br>

      <taglib:myTag myAttributeVIPURL="http://wcmserver.company.example/
         InternetSite_Edit/InternetSite/NewProducts.html" />

</body>
</html>
```

JavaScript example

The following JavaScript example illustrates the use of the different JavaScript expressions for identifying references.

```
<html>
<head>
<title>{VIPTITLE}</title>

<script language="JavaScript">
// JavaScript implementation of the getVIPURL() pattern.
    function getVIPURL(url)
    {
        return url;
    }

    // Leads the browser to the given url.
    function goto(url)
    {
```

```
        location.href = url;
    }

      var url = /*VIPURL*/"http://wcmserver.company.example/
         InternetSite_Edit/InternetSite/ProductOverview.html";

</script>

</head>
<body>

      <a href='javascript:goto(getVIPURL
         ("http://wcmserver.company.example/InternetSite_Edit/
         InternetSite/Products.html"))'>Go to 'Products'</a><br>

      <a onclick='goto(url)'>Go to 'Product Overview'</a><br>
</body>
</html>
```

You can also reference an object by means of its OID. In this case, you precede the OID with the expression `/*VIPOID*/`. The code fragments then look as follows:

- Java: `String oid = /*VIPOID*/"{ID of the object}";`
- JavaScript: `var oid = /*VIPOID*/"{ID of the object}";`

Special case: CSS document

There are different ways of specifying URLs in CSS documents. To enable the WCM system to identify a URL as a reference, you can precede the URL with the expression `/*VIPURL*/`. Please note that the URL must be separated from the right bracket by a `'` (single quote), `"` (double quote), or a space.

Examples

This reference will not be recognized:

```
BODY { background: url(/*VIPURL*/blue.gif) }
```

The following references will be recognized:

```
BODY { background: url(/*VIPURL*/blue.gif ) }
BODY { background: url(/*VIPURL*/"blue.gif") }
BODY { background: url(/*VIPURL*/'blue.gif') }
```

Deleting References

Objective

You want to delete references of an object.

> **Note:** In the References dialog box, you can only delete references that are marked as *NEW*. All other references are already entered in the object content and can only be deleted by making the necessary changes in the content (see section “Editing Object Content” starting on page 178).

Prerequisites

- The reference is labeled *NEW*.
- The object has the status “changed”, “rejected”, “released”, or “delayed release”.
- You have the access rights “Read”, “Change object”, and “Change metadata” for the object.

Procedure

1. Change to the Edit view of the website.
2. In the navigation area, mark the object from which you want to delete a reference.
3. Choose *View* → *References* to open the References dialog box.

4. On the *Referenced objects* tab, mark the reference that you want to delete.
5. Choose *Remove reference* on the context menu.

   The reference is deleted in the WCM system. Since the reference has not yet been entered in the content of the referencing object, this completes the deletion process.

Editing Access Rights

> **Note:** For viewing the Access rights dialog box, you require the functional area “Access rights dialog”. To check your functional areas, choose *Tools* → *Login info*.

For general information on the access rights, see section “Access Rights” starting on page 50.

In this section, the various basic functions for setting the access rights for an object are described. It is explained how to

- add a principal to the access control list of an object, thus granting the principal access to the object, see the following section
- configure access to an object for a principal, see section “Setting the Access Rights of a Principal” starting on page 220
- remove a principal from the access control list of an object, thus denying the principal access to the object, see section “Removing a Principal from the Access Control List” starting on page 222
- configure access to an object for all users of the website, see section “Setting Access Rights for “All” Users” starting on page 223
- configure access for the Production view, see section “Setting Access Rights for the Production View” starting on page 225

- undo access settings you made at object level, see section “Resetting All Object-Specific Access Settings” starting on page 226

You can only change the access settings of an object if you have the access rights “Read”, “Change rights”, and “Change metadata” for the object. Changes to access rights – like all object changes – are subject to the normal staging, i.e. the change must be submitted to Quality Assurance to be checked and released. If Quality Assurance rejects the changes to the access rights, the previous access rights settings are restored.

Notes on changing access rights

- There must be **at least one** principal (user, group, role, or the entry “World”) in the access control list. The last remaining principal cannot be deleted.
- There must be **at least one** principal with the access right “Change rights” in the access control list. If this right is assigned to only one principal, it cannot be withdrawn and the respective principal cannot be deleted.

Adding a Principal to the Access Control List

Objective

You want to add a user (or another entry) to the access control list.

Prerequisites

- The object has the status “changed”, “rejected”, “released”, or “delayed release”.
- You have the access rights “Read”, “Change rights”, and “Change metadata” for the object.

Procedure

1. Change to the Edit view of the website.
2. In the navigation area, mark the object whose access control list you want to edit.
3. Choose *View* → *Access rights*.

   The Access rights dialog box opens.

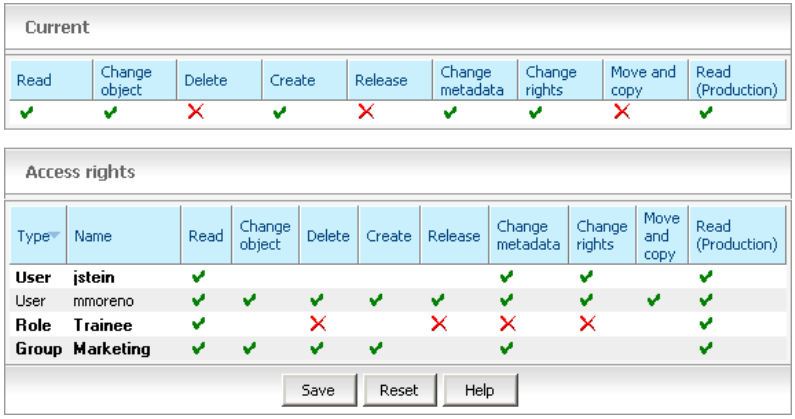


Fig. 71 – The Access rights dialog box

Notes:

The groups and roles to which the logged-in user is assigned are emphasized in bold type in the *Type* and *Name* columns. The access rights in the *Current* section result from the combination of the access rights for the principals shown in bold.

If the *Use access rights of parent topic* button is displayed, the object has an access control list of its own. In this case, the object has not inherited the access settings of the parent topic.

4. Open the context menu of the dialog box by right-clicking the Access rights dialog box.

   The context menu offers commands for the different principal types (users, group, group-role, role, or World).

5. Choose the appropriate command (here: *Add user*).

   The *Add principals* dialog box opens.

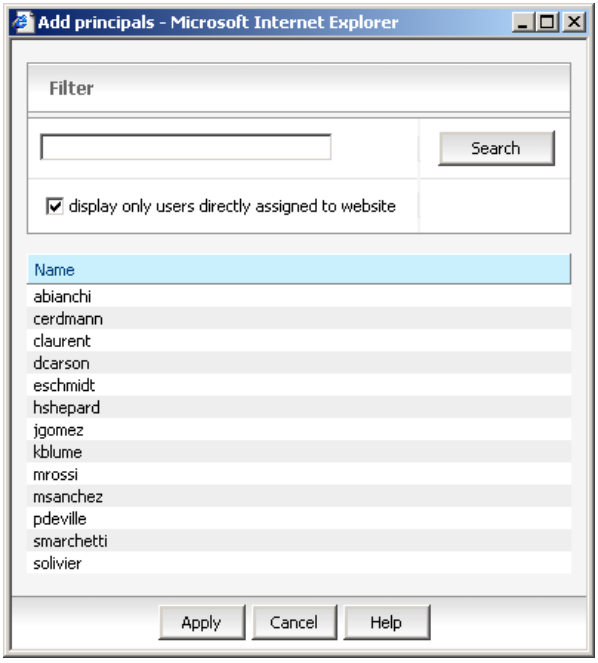


Fig. 72 – Adding a principal to the access control list

6. In the selection list, mark the principal you want to add to the access control list. You can only select individual principals.

7. Click the *Apply* button.

   The principal is inserted in the access control list. This activates the default object rights that are set for this principal in the Admin client. You can change this setting.

8. Click the *Save* button in the Access rights dialog box.

9. Submit the object with the extended access control list to Quality Assurance. To do so, choose *Object* → *Submit* or click the appropriate icon.

   Icon for submitting an object

Searching for principals

It is possible to search for principals. This is especially useful if the selection list is long or if you want to add a user to the access control list who is assigned to the website implicitly as a result of the assignment to groups and roles.

To search for a principal:

1. In the *Filter* field, enter the first letter(s) of the principal that you want to find. If you also want to search for implicitly assigned users, clear the check box *display only users directly assigned to the website*.

   > **Note:** If you want to add a group-role, you can enter two search expressions separated by a space. The first expression is used for searching the group, the second for searching the role.

2. Click the *Search* button.

   The selection list shows only those principals that match the search criterion entered.

3. From the restricted selection list, select the principal you want to add to the access control list.
4. To display the full selection list again, clear the search criterion from the field and click the *Search* button.

Setting the Access Rights of a Principal

Objective

You want to set the access rights for an object.

> **Note:** Please note that the access settings you make for a topic are valid for all subordinate objects of the topic for which no separate access rights have been set.

Prerequisites

- The object has the status “changed”, “rejected”, “released”, or “delayed release”.
- You have the access rights “Read”, “Change rights”, and “Change metadata” for the object.

Procedure

1. Change to the Edit view of the website.
2. In the navigation area, mark the object whose access control list you want to edit.
3. Choose *View* → *Access rights*.
4. Mark the principal entry whose access rights you want to edit.
5. Click the right mouse button → *Edit access rights*.

   The *Edit access rights* dialog box opens.

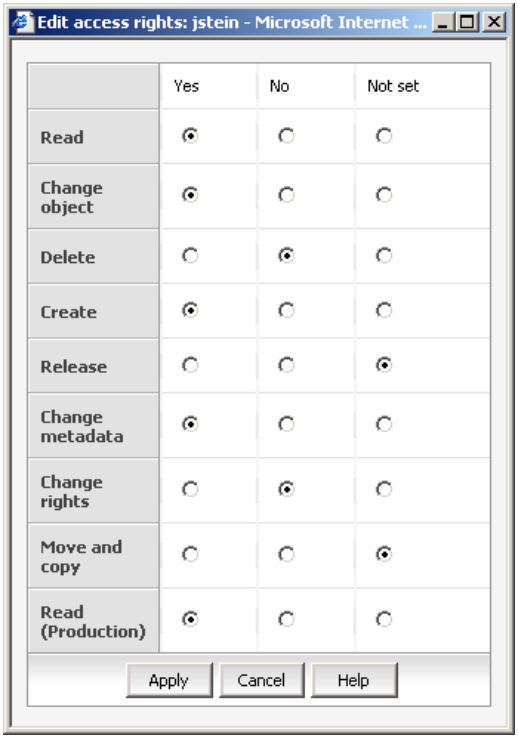


Fig. 73 – Editing access rights

6. Click the desired radio button for the individual rights.
   - *Yes:* The right is explicitely allowed.
   - *No:* The right is explicitely forbidden.
   - *Not set:* The right is not determined.

> **Note:** Access rights are also dependent on group-specific and/or role-specific entries (see section “Priority Rules” on page 57).

7. To apply the changes to the Access rights dialog box, click the *Apply* button.

8. Click the *Save* button in the Access rights dialog box.
9. Submit the object with the edited access control list to Quality Assurance. To do so, choose *Object* → *Submit* or click the appropriate icon.

Icon for submitting an object

Removing a Principal from the Access Control List

Objective

You want to remove a user (or other entry) from the access control list.

Prerequisites

- The object has the status “changed”, “rejected”, “released”, or “delayed release”.
- You have the access rights “Read”, “Change rights”, and “Change metadata” for the object.

Procedure

1. Change to the Edit view of the website.
2. In the navigation area, mark the object whose access control list you want to edit.
3. Choose *View* → *Access rights*.
4. Mark the principal entry that you want to remove.
5. Click the right mouse button → *Remove principal*.

   The principal is removed from the access control list.

6. Click the *Save* button.
7. Submit the object with the changed access control list to Quality Assurance. To do so, choose *Object* → *Submit* or click the appropriate icon.

Icon for submitting an object

Setting Access Rights for “All” Users

Objective

You want control the access to an object (e.g. the right “Change object”) for “all” users.

> **Note:** The principal “World” is a group containing all users. Thus, when it comes to interpreting access rights, the usual priority rules apply to “World” (see section “Priority Rules” on page 57). If you explicitly deny certain rights to the principal “World”, this prohibition can only be overruled by an individual permission. An explicit group-specific or role-specific permission, however, does not overrule the prohibition.

Prerequisites

- The object has the status “changed”, “rejected”, “released”, or “delayed release”.
- You have the access rights “Read”, “Change rights”, and “Change metadata” for the object.

Procedure

1. Change to the Edit view of the website.
2. In the navigation area, mark the object whose access control list you want to edit.
3. Choose *View* → *Access rights*.
4. If the principal “World” is not contained in the access control list, click the right mouse button and choose *Add World* on the context menu.

   The principal “World” is inserted in the access control list.
5. Mark the entry “World” in the access control list and choose *Edit access rights* on the context menu.
6. Set the desired access rights for the principal “World”.
7. To apply the changes to the Access rights dialog box, click the *Apply* button.
8. Click the *Save* button in the Access rights dialog box.
9. Submit the object with the changed access control list to Quality Assurance. To do so, choose *Object* → *Submit* or click the appropriate icon.

Icon for submitting an object

Setting Access Rights for the Production View

In Livelink WCM Server, you can control access to object versions published in the Production view. For the access rights to be considered, the access control component Secure Access or a dynamic, InSite Editing, or WebDAV deployment system must be used (for more information, refer to the Livelink WCM Server Administrator Manual).

- **Unlimited read access in the Production view**

  Read access in the Production view can be permitted on a general basis, i.e. without login. Make sure that the access right “Read (Production)” for the special principal “World” is set to permitted. The object can thus be read by any user who can access the Production view. If the Production view is accessible on the Internet, for example, all Internet users can open the object in their browsers without logging in.

- **Limited read access in the Production view**

  Access to the object versions published in the Production view can be limited, e.g. restricted to certain user groups. In this case, the access right “Read (Production)” is not allowed for the entry “World”. It is only granted to the desired users, groups, or roles. This means the object can only be read by the authorized principals after they have logged in.

Here, too, the access settings are in principle inherited by child objects. Only a change in the access settings for an individual object interrupts the chain of inheritance. Thus, if you permit unrestricted access in the Production view to a particular topic in your website, this access setting also applies to all child objects for which no individual settings have been made (see section “Inheritance of Access Rights” on page 55).

Note: When a website is created, the access right “Read (Production)” is granted by default to the special principal “World”. This setting thus represents a general permission to access the Production view. However, the general rule that users may only access an object if they have the necessary access rights also applies to the Production view. So you can prohibit or restrict this general access at any time by changing the access rights.

Resetting All Object-Specific Access Settings

According to the rules of inheritance, each object inherits the access settings of the parent topic. The settings of the topic may also be inherited. As long as access to the object is regulated satisfactorily by the settings for the topic, the object does not need any individual access settings (see section “Inheritance of Access Rights” on page 55).

If the inherited access rights of an object are changed, the chain of inheritance is interrupted. The child object no longer inherits changes to the access settings of the parent topic.

If necessary, you can discard changes in an object’s access settings and thereby reintegrate the object in the chain of inheritance. Proceed as follows:

1. Change to the Edit view of the website.
2. In the navigation area, mark the object whose access control list you want to edit.
3. Choose *View* → *Access rights*.
4. Click the *Use access rights of topic* button. If this object has already inherited the access settings of the parent topic, this button is not displayed.

Important! Clicking the *Use access rights of topic* button deletes all object-specific access settings **immediately**.

Deleting Objects

In this section, you learn which functions are required for deleting WCM objects. The following procedures are described:

- deleting an object that has already been released at least once, see the following section
- deleting an object that has never been released, see section “Deleting an Object not yet Released” starting on page 229
- destroying an object, see section “Destroying a Released Object” starting on page 230
- rejecting deletion of an object, see section “Rejecting the Deletion of a Released Object” starting on page 231
- removing a published object from the Production view, see section “Removing a Page from the Production View” starting on page 232

Deleting a Released Object

Objective

You want to delete a released object.

Prerequisites

- The object has neither the status “submitted” nor “checked out”. You can only delete a topic if none of its subordinate objects have the status “submitted” or “checked out”.

- The object has been released at least once by Quality Assurance, i.e. it has a major version number of at least 1.
- You can only delete a template if the website contains no objects that use this template. To check this, choose *View* → *Template structure*.
- You have the access rights “Read” and “Delete” for the object.

Procedure

1. Change to the Edit view of the website.
2. In the navigation area, mark the object that you want to delete.
3. Choose *Object* → *Delete* or click the appropriate icon.

   Icon for deleting an object

   If the Content client is configured accordingly, an E-mail dialog box opens.
4. In the E-mail dialog box, make the desired entries and click the *OK* button.
5. Confirm the security prompt by clicking the *Yes* button.

   The object is submitted to Quality Assurance for deletion.

Important! If you delete a topic, all subordinate objects will also be deleted. Child objects that only exist in the Edit view (major version number 0) are permanently deleted as soon as the topic is deleted.

Deleting an Object not yet Released

Objective

You want to delete an object that has never been released by Quality Assurance. The object has the major version number 0. In this case, the object is deleted directly in the Edit view without confirmation by Quality Assurance.

Prerequisites

- The object has the status “changed” or “rejected”.
- The object has not been released by Quality Assurance yet, i.e. it has the major version number 0.
- You have the access rights “Read” and “Delete” for the object.

Procedure

1. Change to the Edit view of the website.
2. In the navigation area, mark the object that you want to delete.
3. Choose *Object* → *Delete* or click the appropriate icon.

   Icon for deleting an object

4. Confirm the security prompt by clicking the *Yes* button.

   The object is irretrievably removed from the WCM system.

Destroying a Released Object

Objective

The version of an object that has already been released once was deleted in the Edit view. In the process of quality assurance, you want to confirm the deletion. This also deletes the object versions in the QA and Production views. The object will be destroyed.

Prerequisites

- The object has the status “deleted”.
- You have the access rights “Read” and “Release” for the object.

Procedure

1. Check whether the object may really be destroyed.
2. In the navigation area, mark the object to be destroyed.
3. Choose *Object* → *Destroy* or click the appropriate icon.

   Icon for destroying an object

   If the Content client is configured accordingly, an E-mail dialog box opens.
4. In the E-mail dialog box, make the desired entries and click the *OK* button.
5. Confirm the security prompt by clicking the *Yes* button.

   The object is now deleted in the WCM system and is no longer available in any view. The object **cannot** be restored.

Rejecting the Deletion of a Released Object

Objective

An object that has already been released once was deleted in the Edit view. You do not agree with this and want to reject the deletion.

Prerequisites

- The object has the status “deleted”.
- You have the access rights “Read” and “Release” for the object.

Procedure

1. In the navigation area, mark the object submitted for deletion.
2. Choose *Object* → *Reject* or click the appropriate icon.

   Icon for rejecting an object

   If the Content client is configured accordingly, an E-mail dialog box opens. In this dialog box, you can reject the deletion of the object and send an explanation to the responsible person(s).
3. Click the *OK* button.

   The object has the status “rejected” and can be checked out and edited.

Removing a Page from the Production View

Objective

You found a serious error in the object version that has already been published in the Production view. You want to remove the faulty page from the Production view without deleting the WCM object itself.

Requirement

You have the access rights “Read” and “Release” for the object.

Procedure

1. In the navigation area, mark the object whose released version you want to remove from the Production view.
2. Choose *Object* → *Remove page*.
3. To check the action, choose *View* → *{name of the Production deployment system}*.

   If the deletion was successful, you get the error “The page cannot be found”. The current version of the object in the Production view has been removed.

Notes:

A page that has been deleted from the Production view by means of the function “Remove page” cannot be restored by means of the function “Regenerate page”. You have to release the current version of the object once again for the version to become available in the Production view.

A page that has been removed from the Production view cannot be found by means of the filter functions of the Content client. Neither is this information saved in the WCM object. Thus, we recommend that you edit such objects directly after performing the “Remove page” function.

Converting Objects

In addition to the familiar method of working where the focus is on individual documents or files, it may be necessary to manage more complex document structures. These structures may consist of a large number of files, e.g. in cases where a conversion program (*converter*) is used to generate numerous HTML files from a single word processor document. The users want to view these complex structures as a single document, since from their point of view, they are a logically coherent object. Although a converter generates many individual HTML pages, they all belong together from a topic point of view. For this reason, the logical connection between the individual HTML pages must not be lost when the object is integrated in the WCM system.

In this section, the following aspects of converting are described:

- process of automatic conversion by means of a converter, see the following section
- differences between converted objects and “normal” objects, see section “Special Properties of Source Objects and Dependent Objects” on page 235
- working with converted objects, see section “Working with Source Objects and Converted Objects” on page 238
- differences between compound objects and “normal” objects, see section “Special Properties of Compound Objects” on page 242

Automatic Conversion

By means of a converter, you can generate complex document structures. That is why the process of automatically converting documents from a word processor or another application into HTML documents is described first. The following diagram illustrates the process.

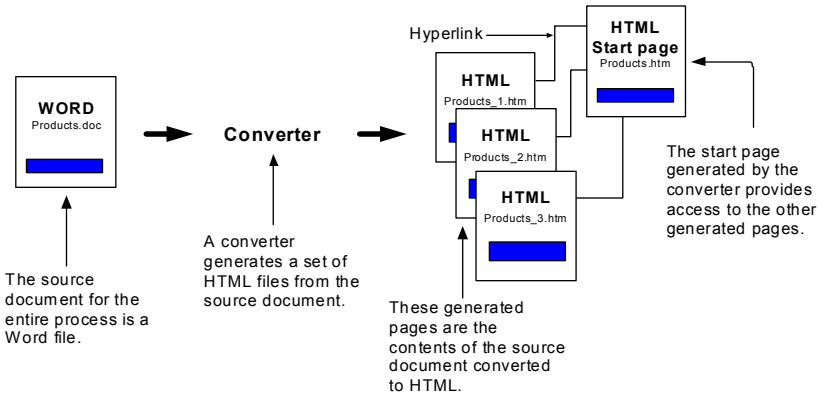


Fig. 74 – Converting a Word file into HTML

The starting point for the conversion is a source document, e.g. a Word or PowerPoint file. The respective WCM object that is based on this document is called *source object*. In our example, we choose a word processor file with the name **Products.doc**. Based on predefined layout rules, the converter converts this document into HTML pages (files). By means of the converter rules, you define where a new HTML page is to be started within the document.

> **Note:** Converters are not supplied with Livelink WCM Server. They are integrated in the WCM system via the Admin client. There your WCM administrator can assign different converters to the different source formats (e.g. **doc**, **ppt**, **xls**).

During the conversion, a set of HTML pages is generated, the so-called *dependent* objects of a source object. The start page is the starting point for navigation within the pages generated by the converter. It has the same name as the source document, but has the file extension **.htm** or **.html**. Thus, in our example the name of the start page is **Products.htm**. The remaining converted HTML pages are named automatically by the converter.

Special Properties of Source Objects and Dependent Objects

The dependent objects of a source object that are generated during conversion basically behave like “normal” WCM objects. However, the logical connection between the source object and its dependent objects requires some special handling.

Staging actions

Staging actions such as “Submit”, ”Release“, or ”Delete“ that you perform on the source object are automatically performed on all dependent objects. This does not apply to the dependent objects that already have the desired status.

Reconverting the source object

You can change the source file. For the changes to become valid for the dependent objects of the source object, the source object must be converted again (“Convert” function). This has the following effects on the individual objects:

- The object status and the version number of the source object remain unchanged.
- The “old” start page is replaced by the regenerated start page. The object gets the status “changed”, the version number is incremented.
- The following applies to the other “old” dependent objects:
  - Objects with a major version number lower than 1 are immediately deleted in the WCM system.
  - Objects with a version number that is higher or equal to 1, get the status “deleted”. They are displayed in addition to the regenerated dependent objects that have the status “changed” and the version number 0.0.1. As soon as the source object is released, the “old” objects will be irretrievably deleted.

Copying the source object

When a source object is copied, the dependent objects are **not** copied. To generate dependent objects for the copy of the source object, convert the copy (“Convert” function).

Copying dependent objects

You may copy dependent objects. However, the copies will not be updated when the source object is reconverted. Instead, new dependent objects will be generated.

Moving the source object

When a source object is moved, the dependent objects are **not** moved. When the source object is reconverted, new dependent objects are generated. The old dependent objects are deleted.

Moving dependent objects

You may move dependent objects. When the source object is converted again, however, the moved objects are deleted and new converted objects are generated at the location in the navigation tree where the source object is located.

Metadata of the source object

The dependent objects of the source object inherit its metadata the first time the source object is converted. After the first conversion, any changes to the metadata of the source object are no longer inherited by the dependent objects. This means that the metadata of the source object and the dependent objects must be maintained separately.

References of the dependent objects

It is possible to change the references of the dependent objects. You can add references to the dependent objects or references from the dependent objects to other objects and make changes to the references between the dependent objects. However, with the exception of the start

page of the source object, all changes get lost when the source object is reconverted.

This results in two different methods for working with a source object and converted objects:

- You create converted objects from a source object only once. All changes and modifications required later are made to the objects generated by the converter. The source object itself is not changed.
- You create converted objects from a source object. All changes that are required subsequently are made in the source object. After checking in the source object, choose the “Convert” function. This automatically generates new dependent objects.

Either one of these working methods prevents changes from being over-written.

Representation of converted objects in the navigation area

In the navigation area, the dependent objects are shown at the same level as the source object (in this case the object “Products” of type “Word document”). The converted objects have been imported into the WCM system as HTML or graphic objects and have the properties of these object types.

You can select each of the dependent objects in the navigation area and display it in the Content client. There is a separate log for each dependent object.

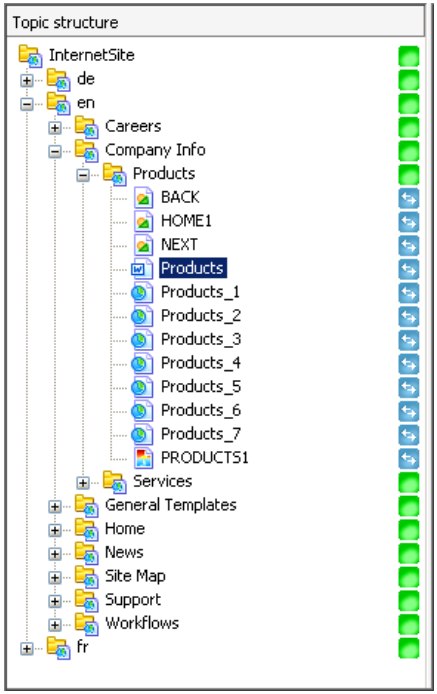


Fig. 75 – Converted objects in the navigation area

Working with Source Objects and Converted Objects

This section explains how to work with converted objects in the Content client. It provides information on the following topics:

- converting a WCM object, see the following section
- editing the source object, see section “Editing the Source Object” on page 240
- submitting the source object, see section “Submitting the Source Object” on page 241

- deleting the source object, see section “Deleting the Source Object” on page 241

Converting an Object

Objective

You want to convert an object into HTML documents.

Prerequisites

- You have the access rights “Read”, “Create”, “Delete”, “Change object”, and “Change Metadata” for the topic in which the source object is located.
- You have the access right “Read” for the source object.
- The source object has the status “changed”, “rejected”, “released”, or “delayed release”.
- You are assigned the functional area “Basic” (or the functional area your WCM administrator has provided for adding objects). To check your functional areas, choose *Tools* → *Login info*.
- In the Admin client, a converter was configured that supports the format of your source file.

Procedure

1. Change to the Edit view of the website.
2. In the navigation area, mark the object that you want to convert.
3. Choose *Object* → *Convert*.

   The converter generates HTML pages that are linked via hyperlinks.

Editing the Source Object

Objective

You want to make changes to a source object. Afterwards, you want to reconvert the source object to generate up-to-date HTML pages.

Prerequisites

- The source object **and** all of its dependent objects have the status “changed”, “rejected”, “released”, or “delayed release”.
- You have the access rights “Read”, “Create”, “Change object”, “Change metadata”, and “Delete” for the source object.

Procedure

Proceed exactly as for editing a “normal” object (see section “Editing Objects” starting on page 177). The only point to be considered is that new dependent objects are created when the changed source object is reconverted. Thus, all changes made to the dependent objects are discarded automatically.

Notes:

A source object, for which the *Direct release* check box is selected, shows a specific behavior: after the source object has been released by Quality Assurance once, it can be released directly by the responsible editor. This also releases all dependent objects with the major version number 0.

The command “Restore old version” in the Log dialog box of the source object **only** refers to the source object, it cannot be applied to its dependent objects. Thus, it is not possible to undo changes made to the dependent objects of a source object.

Submitting the Source Object

Objective

You want to submit a source object to Quality Assurance.

Prerequisites

- The source object has the status “changed”, “rejected”, or “released”.
- None of the dependent objects has the status “checked out”.
- You have the access rights “Read” and “Change object” for the source object.

Procedure

Proceed exactly as for submitting a “normal” object (see section “Submitting an Object” starting on page 247). The only difference you have to note is that submitting a source object automatically submits all its dependent objects as well.

Deleting the Source Object

Objective

You want to delete a source object.

Prerequisites

- The source object has neither the status “submitted” nor “checked out”.
- You have the access rights “Read” and “Delete” for the source object.

Procedure

Proceed exactly as for deleting a “normal” object (see section “Deleting Objects” starting on page 227). The only difference you have to note is that deleting a source object automatically deletes all its dependent objects as well.

Special Properties of Compound Objects

The object type “Compound object” is required for downward compatibility. It was introduced to represent complex document structures generated by means of a converter. When adding a compound object, you specify the source document to be converted as a file. Basically, compound objects behave as “normal” source objects during conversion. There are, however, some differences when working with compound objects. These are described in the following.

Metadata of a compound object

The dependent objects of the compound object inherit its metadata. You may change the metadata of the dependent objects separately from the compound object. However, all changes get lost when the compound object is changed.

Attributes of a compound object

If the compound object and the dependent objects have the same attributes, the dependent objects inherit the values of these attributes from the compound object.

This results in two different working methods for compound objects:

- You add a compound object based on a source document only once. All changes and modifications required later are made to the objects generated by the converter. The source document itself is not changed.
- You add a compound object based on a source document. All required changes are made in the source document. After checking in the source document, choose the “Convert” function. This automatically generates an up-to-date compound object and new dependent objects.

Either one of these working methods prevents changes from being overwritten.

Representation of compound objects in the navigation area

In the navigation area, a compound object is identified by the following icon:

Icon for the object type “Compound object”

Like a topic object, a compound object generates a node in the navigation tree.

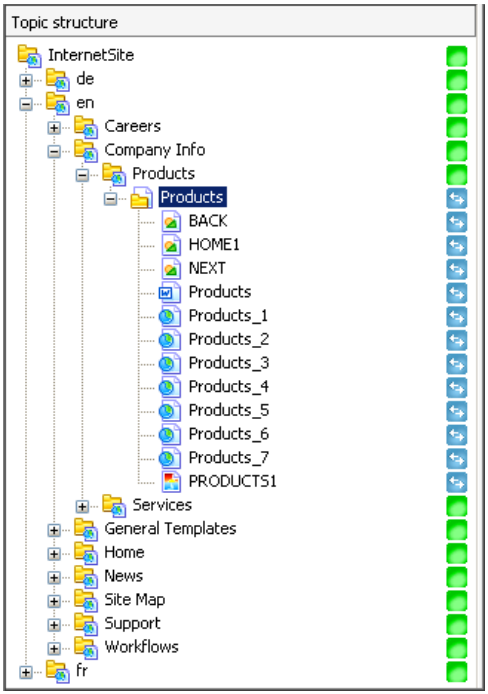


Fig. 76 – The compound object in the navigation area

The object type icon represents the source file the compound object is based on (e.g. a Word or PowerPoint file). Below the compound object, you can see the dependent objects. These have been imported into the WCM system as HTML or graphic objects and have the properties of these object types. The start page is also included in this subtree as an HTML object.

You can select each of the dependent objects in the navigation area and display it in the Content client. There is a separate log for each dependent object.

Compound object in the object filter

The compound object and the dependent objects may be included in a filter's hitlist. You should not perform any actions, such as “Submit” or “Release”, on these objects, as this may lead to errors in the Content client.

CHAPTER 7

Submitting, Releasing, and Rejecting Objects

The staging concept of Livelink WCM Server requires an object to be checked and released by Quality Assurance before it is published in the Production view. This chapter explains how to

- submit new and modified objects to Quality Assurance, see the following section
- release objects for the Production view, see sections “Releasing an Object” on page 251 and “Releasing an Object Directly” on page 253
- reject objects that do not meet the quality standards, see section “Rejecting an Object” on page 255

Submitting an Object

An object must be submitted so that Quality Assurance can check whether it meets the quality standard of the company and can be released. It is possible to undo submitting an object.

Submitting an Object to Quality Assurance

Note: When submitting an object, make sure that you submit the topic first, then the graphics, and finally the HTML pages. You can only submit objects whose parent topic has already been submitted once. If a template is assigned to the object, the template must also have been submitted once.

Objective

You have added an object or edited an existing object. The object is complete and is to be checked by Quality Assurance. You therefore want to submit the object to Quality Assurance.

Prerequisites

- The object has the status “changed”, “rejected”, or “released”.
- You have the access rights “Read”, “Change object”, and/or “Change metadata” for the object.

Procedure

1. Change to the Edit view of the website.
2. In the navigation area, mark the object that you want to submit.
3. Choose *Object* → *Submit* or click the appropriate icon.

   Icon for submitting an object

   The system checks whether there are dependent objects that can be submitted together with the object. Dependent objects are objects that cannot be displayed correctly without the object to be submitted (e.g. images embedded in an HTML page). If there are dependent objects, an according dialog box is displayed.

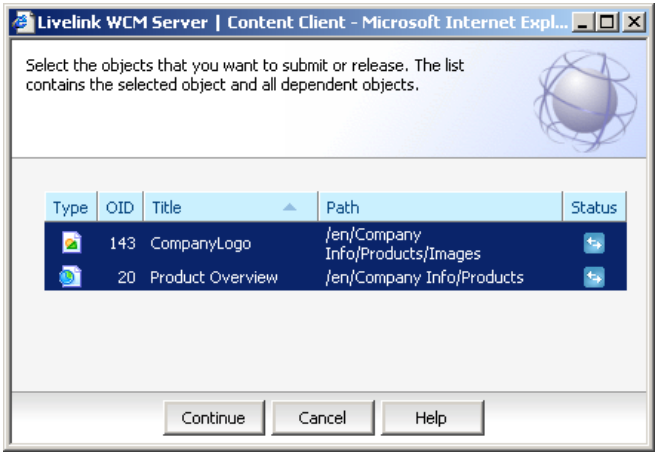


Fig. 77 – Selecting dependent objects when submitting an object

4. Select the dependent objects which are to be submitted together with the object and click the *Next* button.

   If the Content client is configured accordingly, an E-mail dialog box opens. In this dialog box, you can submit the object and send a message to the e-mail addresses saved in the metadata item “E-mail QA”.

5. Click the *OK* button.

   The object has the status “submitted” and can be checked by Quality Assurance.

Undoing Submission of an Object

Objective

You want to undo the action of submitting an object you have edited. This is useful, for example, if you find errors in the submitted object and want to correct them.

Prerequisites

- The object has the status “submitted”.
- You were the last person to submit the object.
- You have the access rights “Read”, “Change object”, and/or “Change metadata” for the object.

Procedure

1. Change to the Edit view of the website.
2. In the navigation area, mark the submitted object.
3. Choose *Object* → *Reject* or click the appropriate icon.

   Icon for rejecting an object

   The object now has the status “rejected”. You can check out the object and edit it again.

Releasing or Rejecting an Object

Livelink WCM Server defines Quality Assurance as the control function between Edit view and Production view. Quality Assurance decides whether an object meets the company's quality standards or needs to be revised.

If the object meets the quality standards, it can be released. This transfers the current version of the object to the Production view. Otherwise the object is rejected and sent back to the responsible editor for correction.

Releasing an Object

Objective

You have completed the quality assurance process for an object and come to a positive conclusion. You now want to transfer the current version of the object to the Production view and make it available to the public.

Prerequisites

- The object has the status "submitted".
- You have the access rights "Read" and "Release" for the object.

Procedure

1. In the navigation area, mark the object that you want to release.
2. Choose *Object* → *Release* or click the appropriate icon.

Icon for releasing an object

The system checks whether there are dependent objects that can be released together with the object. Dependent objects are objects that cannot be displayed correctly without the object to be released (e.g. images embedded in an HTML page). If there are dependent objects, an according dialog box is displayed.

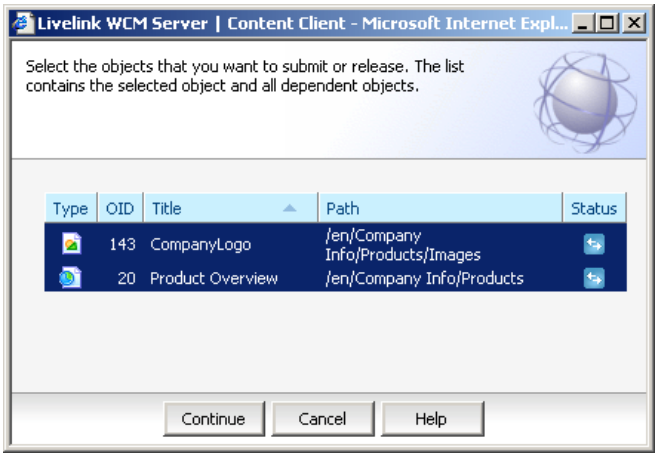


Fig. 78 – Selecting dependent objects when releasing an object

3. Select the dependent objects which are to be released together with the object and click the *Next* button.

   If the Content client is configured accordingly, an E-mail dialog box opens. In this dialog box, you can release the object and send a message to the e-mail address saved in the metadata item “E-mail Release”.

4. Click the *OK* button.

   The object now has the status “released” or “delayed release”. Upon reaching the release date, the released version of the object becomes available in the Production view.

Releasing an Object Directly

Objective

You want to release an object directly without submitting it to Quality Assurance.

Prerequisites

- The object has the status “changed”.
- In the Metadata dialog box of the object, the *Direct release* check box is selected. This setting has already been released once by Quality Assurance.
- You have the access rights “Read”, “Change object”, and/or “Change metadata” for the object.

Procedure

1. In the navigation area, mark the object that you want to release directly.

   If the *Direct release* check box has been selected for the object and the release is possible in the current object status, the “Direct release” function is active in the menu and the toolbar.

2. Choose *Object* → *Direct release* or click the appropriate icon.

   Icon for directly releasing an object

   The system checks whether there are dependent objects that can be released together with the object. Dependent objects are objects that cannot be displayed correctly without the object to be released (e.g. images embedded in an HTML page). If there are dependent objects, an according dialog box is displayed.

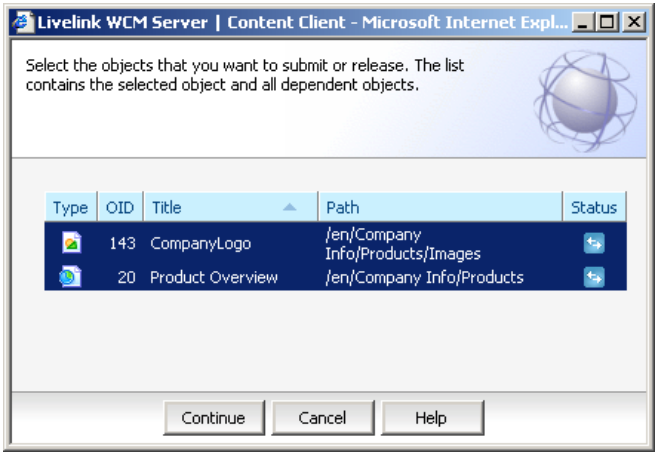


Fig. 79 – Selecting dependent objects when releasing an object directly

3. Select the dependent objects which are to be released together with the object and click the *Next* button.

   If the Content client is configured accordingly, an E-mail dialog box opens. In this dialog box, you can release the object directly and send a message to the e-mail addresses saved in the metadata item “E-mail Release”.

4. Click the *OK* button.

   All the necessary steps will now be performed automatically, i.e. the object is sent to Quality Assurance and automatically released there. Upon reaching the release date, the released version of the object becomes available in the Production view.

Rejecting an Object

Objective

During the quality assurance process, you found errors in an object, and you want to send the object back to the responsible editor for correction.

Prerequisites

- The object has the status “submitted”.
- You have the access rights “Read” and “Release” for the object.

Procedure

1. In the navigation area, mark the object that you want to send back to the responsible editor.
2. Choose *Object* → *Reject* or click the appropriate icon.

   Icon for rejecting an object

   If the Content client is configured accordingly, an E-mail dialog box opens. In this dialog box, you can reject the object and send a message to the e-mail addresses saved in the metadata item “E-mail Edit”.
3. Click the *OK* button.

   The object has the status “rejected” and can be checked out and edited.

CHAPTER 8
Content Workflow

Content Workflow enables you to extend the predefined staging steps by custom workflow steps, e.g. in order to realize editing by several editors or a multi-stage quality assurance. For this purpose, the object type “Workflow” is used. An object of type “Workflow” (workflow object for short) represents a so-called *workflow definition*, which contains the individual workflow steps. Workflow definitions are created and edited with the *Content Workflow Modeler* (workflow modeler for short). In the Content client, workflow objects can be assigned to WCM objects.

For general information on the workflow concept of Livelink WCM Server, refer to section “Workflow” on page 30. In this chapter, you get information on the following workflow aspects:

- elements of a workflow definition, see the following section
- installation of the workflow modeler, see section “Installing the Workflow Modeler” on page 261
- working with the workflow modeler, see section “Working with the Workflow Modeler” on page 262
- creating and editing workflow definitions, see section “Creating and Editing Workflow Definitions” on page 272
- working with WCM objects to which a workflow is assigned, see section “WCM Objects in the Workflow” on page 281

Structure of a Workflow Definition

The workflow definitions created with the workflow modeler are saved in the so-called XML Process Definition Language (XPDL). The format XPDL is XML-based and is the standard defined by the Workflow Management Coalition (WFMC) for interchanging process definitions.

A workflow definition has at least one *starting point* and at least one *end point*. When an object is forwarded in the workflow and reaches an end point, the assignment between WCM object and workflow object is automatically removed. The following types of end points exist:

- Release: When this end point is reached, the WCM object is released. If a delayed release date is set for the object, the object gets the status “delayed release”.
- Stop: When this end point is reached, the assignment between workflow object and WCM object is removed. The status of the WCM object does not change.
- Destroy: When this end point is reached, the WCM object is irretrievably deleted.

A workflow definition may contain any number of *activities* and *transitions*. Each activity is assigned to a principal (user, group, or role) who may edit the object and forward it in the workflow. The three types of activities “Edit”, “QA”, and “Delete” represent the task of the assigned principal.

By means of the transitions, the WCM object is forwarded from one activity to the next. The necessary staging transitions are automatically performed in the background.

When a workflow is assigned to a WCM object, a transition from the starting point to an activity must additionally be selected. The target activity of this transition is called initial activity. The possible transitions depend on the status of the WCM object at the moment the workflow is assigned. The following table provides an overview of the object statuses and the corresponding possible initial activity.

Table 14 – Object statuses and possible initial activity

Object status	Possible activity
rejected	Edit
checked out	none
released	Edit
changed	Edit
deleted	Delete
delayed release	Edit
submitted	QA

Whether the transition from one activity to another activity or to an end point is allowed depends on whether it is possible to perform the respective staging action. For example, the transition from the activity “Delete” to the end point “Release” is not possible. The following table provides an overview of the possible transitions within the workflow and the required access rights.

Table 15 – Possible transitions in the workflow and required access rights

Transition	Required access rights
Edit →	
Edit	Read
QA	Read + Change object
Delete	Read + Delete + Change object + Change metadata
Release	Read + Release + Change object + Change metadata
Destroy	Read + Release + Change object + Change metadata
Stop	Read
QA →	
Edit	Read + Release
QA	Read
Release	Read + Release
Stop	Read
Delete →	
Edit	Read + Release
Delete	Read
Destroy	Read + Release
Stop	Read

Installing the Workflow Modeler

A separate installation of the workflow modeler is only required if you want to use it as a separate application outside the WCM system, e.g. without starting the Content client, or if Java Web Start is not installed on your computer.

> **Note:** For information on the installation requirements for the workflow modeler, refer to the Release Notes for Livelink WCM Server.

To install the workflow modeler:

1. Extract the file **workflowmodeler.zip** from the directory **\workflowmodeler** of the installation CD to a local directory, e.g. **d:\wcm\workflowmodeler**.
2. Open the prompt.
3. Change to the directory selected in step 1.
4. Start the installation by calling the installation script **install.bat**. The following parameters can be used:
   - path to the Java Runtime Environment (JRE):

     If the installation script cannot find a path to a JRE, this parameter must be supplied as the first parameter.

     Example

     ```
     install.bat d:\j2sdk1.4.2\bin
     ```

   - installation directory:

     By means of the parameter `-directory`, you can specify a different installation directory for the workflow modeler. In this case, all files are copied to the specified directory. The start script is created in the new directory.

Example

```
install.bat d:\j2sdk1.4.2\bin
-directory c:\Program Files\WorkflowModeler
```

- languages:

  By means of the parameter `-language`, you can determine the language for the user interface of the workflow modeler. If this parameter is missing, the default language of the operating system is used. Supported languages are English (`en`) and German (`de`).

  Example

  ```
  install.bat d:\j2sdk1.4.2\bin -language en
  ```

After the installation, the workflow modeler can be started by means of the start script **VIPWorkflowModeler.bat** generated during the installation. The start script is located in the workflow modeler installation directory.

Working with the Workflow Modeler

The workflow modeler is a Java application that offers you a graphical user interface for creating and editing workflow definitions. The following gives you a brief introduction to working with the workflow modeler. Described is how

- to start the workflow modeler, see the following section
- to use the menu bar and the toolbar, see section “Menu Bar and Toolbar” on page 265
- to work using the mouse and keyboard, see section “Using Mouse and Keyboard” on page 267

- the individual workflow elements are displayed, see section “Graphical Representation of Workflow Elements” on page 270

Starting the Workflow Modeler

There are different possibilities to start the workflow modeler. The way you choose depends on whether you want to use the Workflow Modeler inside or outside the WCM system and on how your Content client is configured.

Starting the Workflow Modeler Separately from the WCM System

For the workflow modeler to be started separately from the WCM system, it must be installed on your computer. If this is the case, the workflow modeler can be started by double-clicking the start script **VIPWorkflowModeler.bat** generated during the installation. The start script is located in the workflow modeler installation directory.

Starting the Workflow Modeler via the Content Client

Via Java Web Start

For starting the workflow modeler via the Content client, Java Web Start, version 1.4.2, must be installed on your computer. Java Web Start is available for download from Sun Microsystems at http://java.sun.com/products/javawebstart/. If Java Web Start is not installed on your computer, a window with a link for downloading Java Web Start automatically opens the first time the workflow modeler is started.

Additionally, select the check boxes *Use Download applet* and *Use Java Web Start* in the Configuration dialog box (*Tools* → *Configuration*). The workflow modeler will now start automatically when a WCM object of type “Workflow” is checked out. In order to open the workflow modeler without checking out a WCM object, choose *Tools* → *Workflow modeler*.

When starting the workflow modeler for the first time, a security warning appears. You are asked to accept the certificate for the workflow modeler.

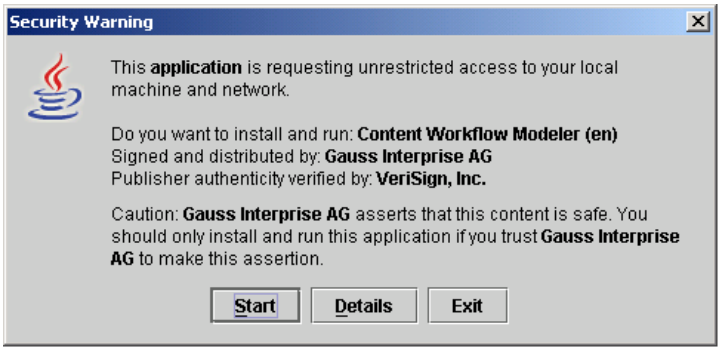

Security Warning
This application is requesting unrestricted access to your local machine and network.
Do you want to install and run: Content Workflow Modeler (en)
Signed and distributed by: Gauss Interprise AG
Publisher authenticity verified by: VeriSign, Inc.
Caution: Gauss Interprise AG asserts that this content is safe. You should only install and run this application if you trust Gauss Interprise AG to make this assertion.
Start
Details
Exit

Fig. 80 – Security warning when starting the workflow modeler

You have the following options:

- *Start:* Click this button to confirm the security prompt. This is necessary for working with the workflow modeler.
- *Details:* Click this button to request more information about the certificate.
- *Exit:* If you do not confirm the prompt, you cannot work with the workflow modeler.

Via an application class

If you do not want to use Java Web Start, you can define the start script **VIPWorkflowModeler.bat**, which is located in the workflow modeler installation directory, as the application for XPDL files in the Configuration dialog box. This presupposes that the workflow modeler is installed on your computer. The workflow modeler will start automatically when an object of type “Workflow” is checked out.

Menu Bar and Toolbar

The menu bar of the workflow modeler provides functions for editing workflow definitions. Additionally, a toolbar offers access to the most important functions. The following table gives you an overview of the commands in the menu bar and the corresponding icons.

Table 16 – Commands in the menu bar of the workflow modeler

Command	Icon	Function
File menu →		
New		Opens a new, empty workflow definition. An already open workflow definition will be closed.
Open		Opens an existing workflow definition. An already open workflow definition will be closed.
Close		Closes the current workflow definition
Reload		Reloads the current workflow definition from the file system. All unsaved changes will be lost.
Save		Saves the current workflow definition
Save as		Saves the current workflow definition in the file system. A dialog box for selecting the storage location opens.
Export as PNG image		Saves the current workflow definition in PNG format
Properties		Opens a dialog box for entering a title and a description for the workflow definition
Page setup		Opens a dialog box for configuring the page settings for printing the workflow definition

Command	Icon	Function
Print		Printing the graphical representation of workflow definitions
Exit		Closes the current workflow definition and quits the workflow modeler
Edit menu →		
Add *“Start”*		Adds a new starting point
Add activity		Adds a new activity
Add “Release”		Adds an end point of type “Release”
Add “Destroy”		Adds an end point of type “Destroy”
Add “Stop”		Adds an end point of type “Stop”
View menu →		
Original size		Restores the original size of the display
Zoom in		Increases magnification of the display
Zoom out		Decreases magnification of the display

Command	Icon	Function
Help menu →		
Help topics		Opens the online help of the workflow modeler in a separate window **Note:** You can also open the online help by pressing the F1 key.
About		Displays information about the WCM system used

Using Mouse and Keyboard

The following operations in the workflow modeler can only be executed via the mouse or keyboard:

- moving activities, starting points, and end points, see the following section
- changing the size of activities, starting points, and end points, see section “Changing the Size of Activities, Starting Points, and End Points” on page 268
- adding transitions, see section “Adding Transitions” on page 268
- moving transitions, see section “Moving Transitions” on page 269
- deleting workflow elements, see section “Deleting Workflow Elements” on page 270

Moving Activities, Starting Points, and End Points

To move an activity, a starting point, or an end point:

1. Make sure that the transition mode is switched off. If necessary, click the icon to deactivate the transition mode.

2. Mark the element to be moved with the left mouse button.

   The element is highlighted by an additional frame.

3. While holding the left mouse button down, move the element to the desired location.

Changing the Size of Activities, Starting Points, and End Points

To change the size of an activity, a starting point, or an end point:

1. Make sure that the transition mode is switched off. If necessary, click the icon to deactivate the transition mode.
2. Mark the desired element with the left mouse button.

   The element is highlighted by an additional frame.

3. Place the mouse pointer on a small box within this frame.

   The mouse pointer changes to a double-sided arrow.

4. While holding the mouse button down, drag the element to the desired size.

Adding Transitions

Note: Not all transitions are possible. Also, the principal assigned to the source activity must have the required access rights for the transition (see table “Possible transitions in the workflow and required access rights” on page 260). The workflow modeler does not check the access rights.

To add a transition:

1. To activate the transition mode, click the icon .
2. Place the mouse pointer on the activity or the starting point to be used as the source of the transition.

The mouse pointer changes to a hand.

3. While holding the left mouse button down, drag the displayed line to the activity or end point that is to be the target of the transition until this activity or end point is highlighted by an additional frame.
4. Release the mouse button.
5. To deactivate the transition mode, click the icon .

Note: It is also possible to add transitions without activating the transition mode. In this case, you must keep the SHIFT key pressed during the entire process of adding the transition.

Moving Transitions

It is not possible to move a transition as a whole. The positions of the transition's ends must be changed separately.

To move a transition:

1. Make sure that the transition mode is switched off. If necessary, click the icon to deactivate the transition mode.
2. Mark the desired transition with the left mouse button.

   The transition is highlighted in green.
3. Place the mouse pointer on the end of the transition that you want to move.

   The mouse pointer changes to a cross.
4. While holding the left mouse button down, move the end of the transition to the desired location at an activity, a starting point, or an end point.

   When the mouse button is moved over a location to which the end of the transition may be moved, a small square is displayed or the respective starting point or end point is highlighted by a frame.

5. Release the mouse button.
6. If necessary, repeat steps 3 to 5 for the other end of the transition.

Deleting Workflow Elements

> **Note:** When an activity, starting point, or end point is deleted, all related transitions are removed, too.

To delete an activity, a starting point, or an end point:

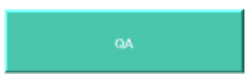
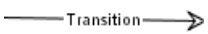
1. Make sure that the transition mode is switched off. If necessary, click the icon to deactivate the transition mode.
2. Mark the desired element with the left mouse button.
3. Press the DEL key.

   The selected element is deleted.

Graphical Representation of Workflow Elements

In the workflow modeler, transitions, activities, starting points, and end points have a graphical representation. The following table shows the representation of the individual workflow elements in the workflow modeler.

Table 17 – Graphical representation of workflow elements

Workflow element	Graphical representation
Starting point	
End point “Release”	
End point “Stop”	
End point “Destroy”	
Activity “Edit”	
Activity “QA”	
Activity “Delete”	
Transition	

Creating and Editing Workflow Definitions

This section describes how to

- create a workflow definition, see the following section
- edit existing workflow definitions, see section “Editing a Workflow Definition” on page 278
- delete existing workflow definitions, see section “Deleting a Workflow Definition” on page 280

Creating a Workflow Definition

This section describes how to create a workflow definition. The procedure depends on whether you start the workflow modeler via the Content client or as a separate application.

Creating a Workflow Definition via the Content Client

Prerequisites

- You have the access rights “Read” and “Create” for the topic to which you want to add the WCM object of type “Workflow”.
- You are assigned the functional area “Workflow” (or the functional area your WCM administrator has provided for adding workflow objects). To check your functional areas, choose *Tools* → *Login info*.

Step 1 – Add workflow object

Add a WCM object of type “Workflow”.

Step 2 – Check out the workflow object

Check out the new workflow object.

The workflow modeler opens. You can now start to create the workflow definition. The new workflow definition already contains a starting point. If you do not want to add additional starting points, proceed with step 4.

Step 3 – Add starting points

1. Make sure that the transition mode is switched off. If necessary, click the icon to deactivate the transition mode.
2. Choose *Edit* → *Add "Start"* or click the appropriate icon.

   Icon for adding a starting point

   The new starting point is inserted in the top left window pane.
3. Mark the new starting point with the mouse and – while holding the left mouse button down – move it to the desired location.
4. Double-click the starting point.

   The *Edit activity* dialog box contains the following information:

   - *Name:* Shows the name of the starting point ("Start"). You cannot change this entry.
   - *Activity type:* This field shows the type of the activity. You cannot change this entry.
   - *Description:* In this field, you can enter additional information on the starting point.
5. Click the *OK* button.
6. To add additional starting points, repeat steps 2 to 5.

Step 4 – Add activities

1. Make sure that the transition mode is switched off. If necessary, click the icon to deactivate the transition mode.

2. Choose *Edit* → *Add activity* or click the appropriate icon.

   Icon for adding an activity

   The *New activity* dialog box opens.

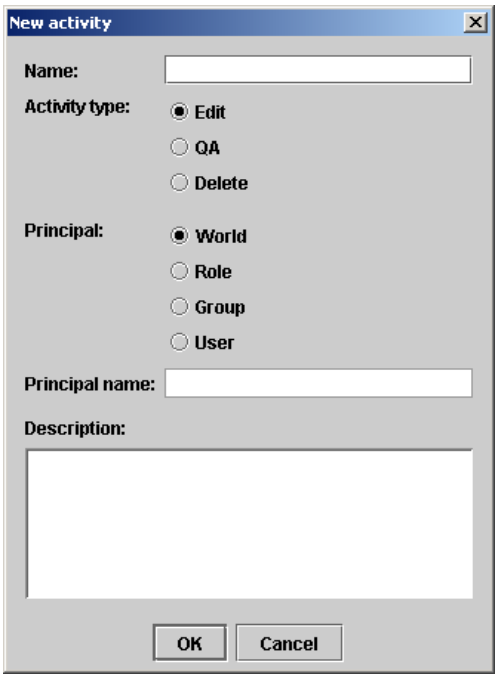


Fig. 81 – Dialog box when adding an activity

You can make the following settings:

- *Name:* Enter the name for the activity.
- *Activity type:* Select the type of the activity.
- *Principal:* Specify the type of the principal to be assigned to the activity.

- *Principal name:* Enter the ID of the user or the name of the group or role that you want to assign to the activity. If you have clicked the radio button *World* under *Principal*, no entry is required here.

  Notes:

  If the user data of your WCM system is managed by means of the LDAP directory service Microsoft Active Directory, enter the user name instead of the user ID. For further information, consult your WCM administrator.

  Note that the assigned principal must have the access rights required for editing the object within the activity.

- *Description:* Enter additional information on the activity.

3. Click the *OK* button.

   The new activity is inserted in the top left window pane.

4. Mark the new activity with the mouse and – while holding the left mouse button down – move it to the desired location.

5. To add more activities, repeat steps 2 to 4.

Step 5 – Add end points

1. Make sure that the transition mode is switched off. If necessary, click the icon to deactivate the transition mode.

2. Choose *Edit* → *Add "Release"*, *Add "Destroy"*, or *Add "Stop"* or click one of the appropriate icons.

   Icon for adding the end point "Release"

   Icon for adding the end point "Destroy"

   Icon for adding the end point "Stop"

The new end point is inserted in the top left window pane.

3. Mark the new end point with the mouse and – while holding the left mouse button down – move it to the desired location.
4. Double-click the end point.

   The *Edit activity* dialog box contains the following information:

   - *Name:* This field shows the name of the end point. You cannot change this entry.
   - *Activity type:* This field shows the type of the activity. You cannot change this entry.
   - *Description:* In this field, you can enter additional information on the end point.
5. Click the *OK* button.
6. To add more end points, repeat steps 2 to 5.

Step 6 – Define transitions

1. Add a transition.
2. Double-click the transition.
3. In the *Edit transition* dialog box enter the name and a description for the transition.
4. Click the *OK* button.
5. To define more transitions, repeat steps 1 to 3.

Step 7 – Define workflow properties

1. For this purpose, choose *File* → *Properties*.

   The *Edit workflow properties* dialog box opens.
2. Enter a name and a description for the workflow definition.
3. Click the *OK* button.

Step 8 – Complete workflow definition

Save the workflow definition and close the workflow modeler. The workflow object now passes through the normal staging steps of Livelink WCM Server, i.e. it must be submitted to Quality Assurance, which either releases or rejects it.

After the workflow object has been released, it can be assigned to other WCM objects.

Creating a Workflow Definition Without the Content Client

Step 1 – Start workflow modeler

Start the workflow modeler by double-clicking the file **VIPWorkflowModeler.bat**, which is located in the workflow modeler installation directory.

Step 2 – Create workflow definition

Choose *File* → *New* or click the appropriate icon.

Icon for creating a workflow definition

Step 3 – Edit workflow definition

The following steps correspond to steps 3 to 7 when creating a workflow definition via the Content client.

Step 4 – Save the workflow definition

1. Choose *File* → *Save* or click the appropriate icon.

   Icon for saving a workflow definition

   The *Save workflow definition* dialog box opens.

2. From the *Save in* drop-down list, select the desired storage location. In the *File Name* field, enter the name for the workflow definition file (XPDL format).
3. Click the *OK* button.

   The workflow definition is saved.

Step 5 – Add workflow object

If you want to use the new workflow definition in the WCM system, add a WCM object of type “Workflow”. The following requirements must be met:

- You have the access rights “Read” and “Create” for the topic to which you want to add the WCM object of type “Workflow”.
- You are assigned the functional area “Workflow” (or the functional area your WCM administrator has provided for adding workflow objects). To check your functional areas, choose *Tools* → *Login info*.

On the basis of the XPDL file saved in step 4, add a WCM object of type “Workflow”.

The workflow object now passes through the normal staging steps of Livelink WCM Server, i.e. it must be submitted to Quality Assurance, which either releases or rejects it.

After the workflow object has been released, it can be assigned to other WCM objects.

Editing a Workflow Definition

Via the Content client

1. Check out the workflow object.

   The workflow modeler opens.

2. Edit the workflow definition. You can:
   - add workflow elements, see section “Creating a Workflow Definition” on page 272 and “Adding Transitions” on page 268
   - delete workflow elements, see section “Deleting Workflow Elements” on page 270
   - move workflow elements, see sections “Moving Activities, Starting Points, and End Points” on page 267 and “Moving Transitions” on page 269
   - double-click a workflow element to edit its properties
3. Save your changes and close the workflow modeler.

> **Notes:**
>
> When a workflow is assigned to a WCM object, the latest released version is always used. Thus, the respective workflow object must be released first before the changed workflow definition can be used.
>
> The changed workflow definition does not affect WCM objects to which the workflow has already been assigned. These objects pass through the old version of the workflow.

Via the workflow modeler as a separate application

1. Start the workflow modeler by double-clicking the file **VIPWorkflowModeler.bat**, which is located in the workflow modeler installation directory.
2. Open the workflow definition via *File* → *Open* or click the appropriate icon.

Icon for opening a workflow definition

The *Open workflow definition* dialog box opens.

3. Select the desired workflow definition and confirm by clicking the *Open* button.

   The selected workflow definition opens.

4. Edit the workflow definition. You can:

   - add workflow elements, see section “Creating a Workflow Definition” on page 272 and “Adding Transitions” on page 268
   - delete workflow elements, see section “Deleting Workflow Elements” on page 270
   - move workflow elements, see sections “Moving Activities, Starting Points, and End Points” on page 267 and “Moving Transitions” on page 269
   - double-click a workflow element to edit its properties

5. Save your changes and close the workflow modeler.

Note: The changed workflow definition does not automatically become available in the WCM system. For this purpose, you must add a WCM object of type “Workflow” on the basis of the changed XPDL file or change an existing workflow object.

Deleting a Workflow Definition

Via the Content client

To delete a workflow definition, you must only delete the respective WCM object of type “Workflow” in the Content client.

If you want to delete the WCM object only without deleting the workflow definition, proceed as follows:

1. Check out the workflow object to be deleted.

   The workflow modeler opens.

2. Save the workflow definition in the file system. Choose *File* → *Save as* or click the appropriate icon.

   Icon for saving a workflow definition in the file system

3. Close the workflow modeler and delete the respective WCM object.

Via the workflow modeler as a separate application

Delete the respective XPDL file in the file system. Please note that in this case, the workflow definition is not deleted in the WCM system. In order to delete a workflow definition from the WCM system, proceed as described above.

WCM Objects in the Workflow

In the Content client, you can assign to WCM objects the workflow definitions that you have created by means of the workflow modeler. When editing the object, the user will select one of the predefined workflow steps. The normal staging actions are no longer available for this object.

In this section, you get information on the following aspects:

- assigning a workflow to a WCM object, see the following section
- removing the assignment of a workflow to a WCM object, see section “Removing a Workflow Assignment” on page 284
- editing WCM objects to which a workflow is assigned, see section “WCM Objects in the Workflow” on page 285
- forwarding an object through an entire workflow on the basis of an application example, see section “Application Example” on page 287

Assigning a Workflow

You can assign a workflow to a WCM object directly when adding the WCM object or at any time later. In the following, both options will be described.

Assigning a Workflow when Adding the Object

Prerequisites

- You have the access right “Read” for the workflow object that you want to assign to the new WCM object.
- The workflow object has been released at least once by Quality Assurance, i.e. it has a major version number of at least 1.
- You have the access rights “Read” and “Create” for the topic to which you want to add the WCM object.
- You are assigned the functional area “Workflow” and the respective functional area for adding (or the functional area your WCM administrator has provided for this purpose). To check your functional areas, choose *Tools* → *Login info*.

Procedure

In the first dialog box displayed when adding an object, click the workflow that you want to assign to the new object in the *Workflow* drop-down list.

Assigning a Workflow to an Existing Object

Prerequisites

- You have the access right “Read” for the workflow object that you want to assign to the WCM object.
- You are assigned the functional area “Workflow”. To check your functional areas, choose *Tools* → *Login info*.

- The workflow object has been released at least once by Quality Assurance, i.e. it has a major version number of at least 1.
- You have the access right “Read” for the WCM object to which you want to assign the workflow.
- The status of the WCM object allows one of the transitions of the starting point (see table “Object statuses and possible initial activity” on page 259).

Procedure

1. In the navigation area, mark the object to which you want to assign the workflow.
2. Choose *Object* → *Assign workflow* or click the appropriate icon.

   Icon for assigning a workflow

   The *Assign workflow* dialog box opens. This dialog box lists all available workflows.

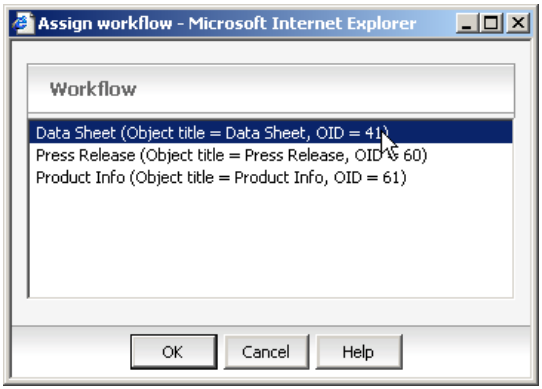

Assign workflow - Microsoft Internet Explorer
Workflow
Data Sheet (Object title = Data Sheet, OID = 41)
Press Release (Object title = Press Release, OID = 60)
Product Info (Object title = Product Info, OID = 61)
OK
Cancel
Help

Fig. 82 – Dialog box for assigning a workflow

3. Select the desired workflow and click the *OK* button.

If the assigned workflow has several starting points or the starting point has several transitions, the *Forward* dialog box opens.

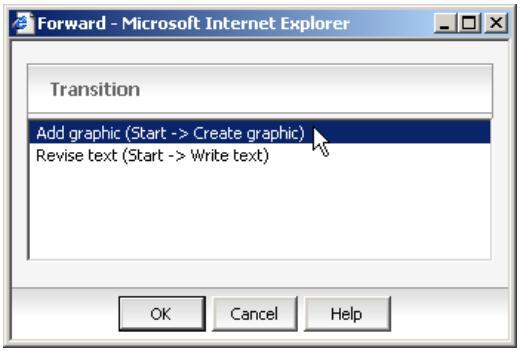


Fig. 83 – Dialog box for selecting a transition

4. Select the desired transition and click the *OK* button.

   The selected workflow is assigned to the current WCM object.

Removing a Workflow Assignment

If the WCM object reaches an end point in the workflow, the assignment between workflow and object is automatically removed. However, it is possible that an object cannot be forwarded in the workflow, e.g. because the principal assigned to the current activity does not have the required access rights. In this case, you must remove the workflow manually.

Objective

You want to remove the assignment between a workflow and a WCM object.

Prerequisites

- You have the access rights “Read” and ”Change rights“ for the WCM object from which you want to remove the workflow. If you assigned the workflow to the WCM object and the object has not yet been forwarded in the workflow, the right “Read” is sufficient.
- You are assigned the functional area “Workflow”. To check your functional areas, choose *Tools* → *Login info*.

Procedure

1. In the navigation area, mark the WCM object from which you want to remove the workflow.
2. Choose *Object* → *Remove workflow assignment*.
3. Confirm the security prompt by clicking the *Yes* button.

   The assignment between WCM object and workflow is removed.

WCM Objects in the Workflow

It is not possible to perform normal staging actions on a WCM object to which a workflow is assigned. The object may only be edited by the principal assigned to the current activity. Depending on the access rights, this principal may check out, edit, check in, and move the object, as well as restore an earlier object version.

Having edited the WCM object, the user forwards the object in the workflow. For this purpose, the user selects a predefined transition. This takes the object to the next activity. If this involves a staging transition, the user must have the required access rights.

Choose *View* → *Workflow* to display information on the assigned workflow.

Objective

You want to edit a WCM object to which a workflow is assigned.

Prerequisites

- You are assigned to the current activity of the object. Choose *View* → *My work list* for an overview of all objects in the workflow that you may currently edit.
- You have the access rights that are required for editing the object and forwarding it in the workflow.

Procedure

1. In the navigation area, mark the object that you want to edit.
2. Check out the object and edit it as usual.
3. When you have finished editing the object, check the object back in to the WCM system.
4. Choose *Object* → *Forward* or click the appropriate icon.

   Icon for forwarding an object in the workflow

   If the Content client is configured accordingly, an E-mail dialog box opens. In this dialog box, you can forward the object and send a message to the principal assigned to the next activity.

   If the current activity has only one transition, the object is forwarded automatically. Otherwise, the *Forward* dialog box opens.
5. Select a transition and confirm by clicking the *OK* button.

   If the Content client is configured accordingly, a Remark dialog box opens.
6. Enter a remark for the log and click the *OK* button.

The object is forwarded to the next activity. If you have selected a transition to an end point, the assignment between object and workflow is removed.

Application Example

For a better understanding, the following example shows the procedure of forwarding an object through a workflow.

> **Note:** The directory **{WCM installation directory}\examples\workflows** contains further example workflows in XPDL format.

Scenario

The group “Marketing” is, among other things, responsible for creating product data sheets. Within the group, there are the roles “Editor”, “Designer”, “QA 1”, and “QA 2”, which have different access rights according to their tasks. The following table shows the roles within the group “Marketing” with the assigned users and access rights, as well as the respective tasks.

Table 18 – The group “Marketing”

Role	User	Access rights	Task
Editor	kblume, pdeville	Read + Create + Change object + Delete	Create product descriptions
Designer	mmoreno, hshepard	Read + Change object	Create graphics for product descriptions
QA 1	dcarson, eschmidt	Read + Release	Check text and graphics
QA 2	mrossi, abianchi	Read + Release	Finally check text and graphics

A workflow for editing data sheets already exists. In the WCM system, this workflow is available as workflow object with the title “DataSheet”. The following figure shows this workflow.

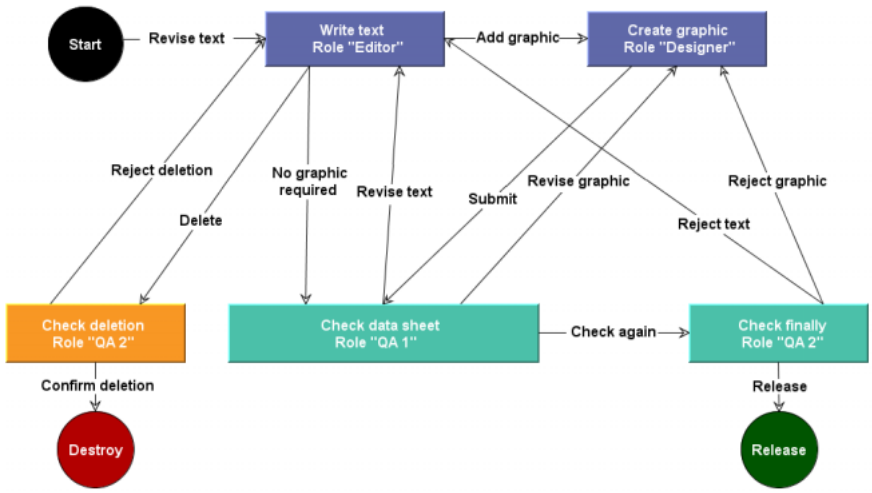


Fig. 84 – Example workflow for editing data sheets

Procedure

1. Product management gives the editor “kblume” the task to revise the data sheet “ProductFeatures”. Kblume assigns the workflow “DataSheet” to the existing and released WCM object.

   The WCM object's status remains “released”. The transition to the first activity “Write text”, to which the role “Editor” is assigned, is performed automatically.

2. By mistake, the editor “pdeville” thinks that the data sheet is no longer required. Thus, he selects the transition “Delete”. In the E-mail dialog box, he enters a reason.

The object “ProductFeatures” is forwarded to the activity “Check deletion”, to which the role “QA 2” is assigned, and gets the status “deleted”.

3. The QA member “abianchi” checks the reason given by “pdeville”. Because the data sheet is still required, she selects the transition “Reject deletion”.

   The WCM object gets back to the activity “Write text” and has the status “rejected”.

4. The editor “kblume” revises the text for the data sheet and selects the transition “Add graphic”. In the E-mail dialog box, he briefly describes what the graphic should look like and where it is to be inserted in the text.

   The object “ProductFeatures” is forwarded to the activity “Create graphic”, to which the role “Designer” is assigned, and gets the status “changed”.

5. The graphics specialist “hshepard” creates the desired graphic and adds it to the data sheet. There is only one transition leading away from the activity “Create graphic”. This transition is performed automatically when “hshepard” forwards the WCM object in the workflow.

   This takes the WCM object to the activity “Check data sheet”, the first stage of quality assurance. The object has the status “submitted”.

6. The QA member “dcarson” checks both text and graphic and approves of them. He forwards the WCM object to the second stage of quality assurance by selecting the transition “Check again”.

   The object “ProductFeatures” remains in the status “submitted” and is now in the activity “Check finally”.

7. The QA member “mrossi” checks the data sheet again. He does not approve of the graphic. Thus, he selects the transition “Reject

graphic” and enters the reason for the rejection in the E-mail dialog box.

The object gets back to the activity “Create graphic” and has the status “rejected”.

8. The graphics specialist “mmoreno” makes the desired changes. The object “ProductFeatures” now has the status “changed”.

   “mmoreno” forwards the object to the first stage of quality assurance. The object gets the status “submitted” and is in the activity “Check data sheet”.

9. The first stage of quality assurance approves of the data sheet. By selecting the transition “Check again”, “eschmidt” forwards the WCM object to the second stage of quality assurance.

   In the second stage of quality assurance, “mrossi” checks the data sheet and is satisfied with the result. Thus, he releases the WCM object by selecting the transition “Release”.

   The object acquires the status “released” and the workflow assignment is automatically removed.

CHAPTER 9
InSite Editing

InSite Editing makes it possible to change or add content directly in a website. A simple click with the mouse during surfing in the Edit or QA view and – according to the WCM rights – context menus with the available edit functions are displayed. This makes it considerably easier to edit content. InSite Editing is also subject to the staging steps of editing, quality assurance, and publication.

Before you can use InSite Editing, you must perform the following steps:

1. Configure the JSP engine properly (responsible: WCM system administrator).

   For detailed information, refer to the Livelink WCM Server Installation Manual.

2. Set up an InSite Editing deployment system for the website (responsible: WCM system administrator).

   For detailed information, refer to the online help of the Admin client.

3. Optional: configure InSite Editing (responsible: web designer).

   For detailed information, refer to section “Configuring InSite Editing” on page 309.

This chapter describes how to work with InSite Editing. This includes:

- introduction to working with InSite Editing, see the following section
- example for editing a website via InSite Editing, see section “Application Example: The HeroLeasing Website” on page 304
- configuring InSite Editing, see section “Configuring InSite Editing” on page 309

Working with InSite Editing

This section contains an introduction to working with InSite Editing. It is described how to

- start InSite Editing, see the following section
- call functions for WCM objects in InSite Editing, see section “Calling Functions in InSite Editing” on page 296
- edit the content of objects via InSite Editing, see section “Editing Object Content via InSite Editing” on page 297
- edit object fragments via InSite Editing, see section “Editing Object Fragments via InSite Editing” on page 298

Note: The responsible web designer may configure InSite Editing and customize it to suit your company's requirements. For this reason, the look and functions of your InSite Editing may differ from the following description.

Starting InSite Editing

To start InSite Editing:

1. Launch your browser.
2. Enter the URL your WCM system administrator configured for InSite Editing in the address bar of your browser (e.g. http://wcmserver.company.example/vipdyn).

   Alternately, you can start InSite Editing by choosing *View* → *{name of the InSite Editing deployment system}* in the Content client.

   If you have not started InSite Editing via the Content client, a dialog box for logging in to the WCM system opens. The Login dialog box of Internet Explorer is shown here as an example.

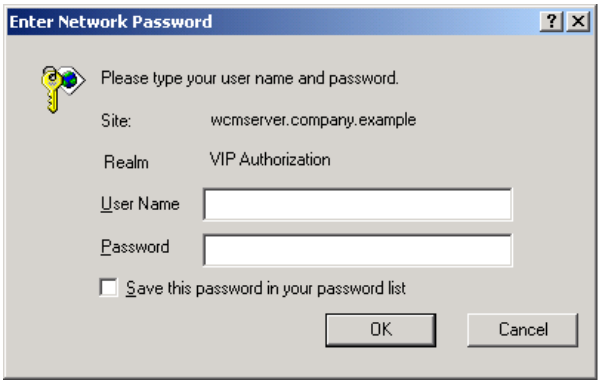


Fig. 85 – Logging in to InSite Editing

3. Enter your user ID for the WCM system and the corresponding password and click the *OK* button.

   This takes you to the Edit view of the website.

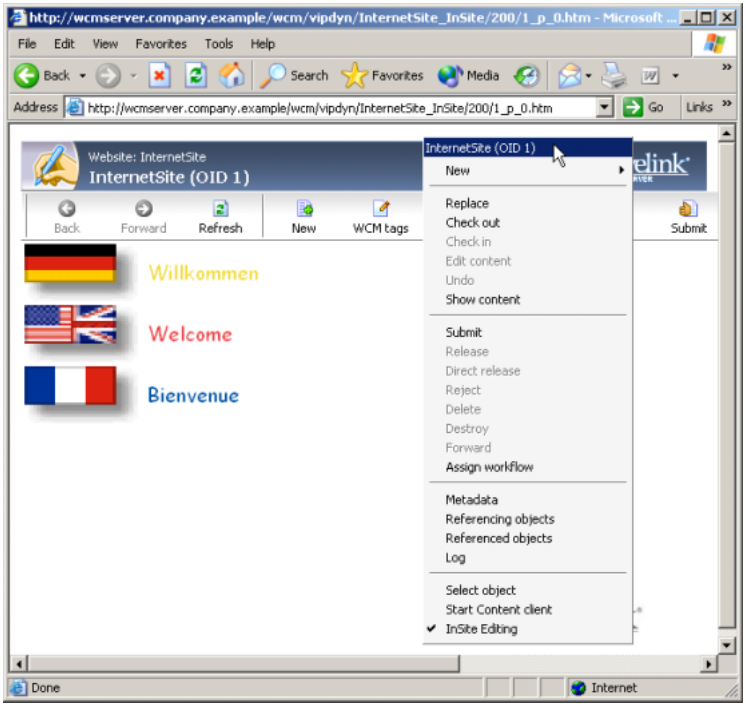


Fig. 86 – InSite Editing

Calling Functions in InSite Editing

In InSite Editing, you use the toolbar or the context menu to access the individual editing functions. Click the right mouse button to open the context menu. The menu items and icons are dynamically modified to suit the current context. Functions that you cannot use for the current object are deactivated (dimmed).

Notes

- Clicking the *Back* button of your browser might lead to JavaScript errors. For this reason, you should use the toolbar of InSite Editing or the *Select object* function on the context menu for navigating.
- InSite Editing and the Content client are **not** synchronized. Thus, changes made by means of InSite Editing do not become automatically visible in a Content client that is open at the same time. To update the view in the client, choose *View* → *Refresh* or press F5.
- If you do not use InSite Editing for a longer period of time, an error message due to a session timeout of the JSP engine or the application server may occur. In this case, restart your browser. Your WCM administrator can modify the timeout value of the JSP engine or the application server.
- You cannot use InSite Editing for frame objects.

Editing Object Content via InSite Editing

InSite Editing enables you to edit the content of objects directly in the website. The following prerequisites must be fulfilled for this purpose:

- The object must be assigned a template in which the `edit` parameter is set for the WCM tag `{VIPCONTENT}` (see chapter 15 “Working with WCM Tags”) and section “The `edit` Parameter” on page 458).
- The object has the status “changed”, “rejected”, “released”, or “delayed release”.
- You have the access rights “Read” and “Change object” for the object.

If you move the mouse over the object’s content, it is highlighted by a red frame. This frame is only displayed if the content can be edited.

To edit the content of the object:

1. Start InSite Editing.
2. Navigate to the object whose content you want to edit.

   The content is highlighted by a red frame.
3. To check out the object, choose *Check out* on the context menu or click the icon in the upper right corner of the frame.

   Depending on the configuration of the Content client, the object is opened in the integrated HTML editor or in the associated editor.
4. Edit the object in the editor.
5. When you have finished editing the object, check the object back in to the WCM system. To do so, choose *Check in* on the context menu or click the icon in the upper right corner of the frame.

   The object has the status “changed”.
6. To submit the object to Quality Assurance, choose *Submit* on the context menu, or click the appropriate icon.

   Icon for submitting an object

Editing Object Fragments via InSite Editing

InSite Editing can also be used for editing information integrated in the page by means of WCM tags directly in the website. This is, for example, especially convenient if the current page contains the content of another WCM object. In this case, you do not have to navigate to the referenced object. You can edit it directly.

Information that can be edited separately via InSite Editing without checking out the object is called *Object fragments*. Each fragment in InSite Editing is represented by a metadata item and can thus be changed via the Metadata dialog box.

Additionally, InSite Editing provides a special dialog box displaying only the information integrated by means of WCM tags. To open the WCM tag dialog box, click the following icon:

The following prerequisites must be fulfilled for editing object fragments via InSite Editing:

- The information is integrated in the page by means of WCM tags (see chapter 15 “Working with WCM Tags”).
- For these WCM tags, the parameter `edit` must be set (see section “The `edit` Parameter” on page 458).

If you move the mouse over a fragment that can be edited with InSite Editing, the fragment is highlighted by a red frame. The fragments are only highlighted if they can be edited. This depends on the object status and the access rights of the logged-in user.

For each fragment, different functions are available via the context menu and a small toolbar. You open the context menu by right-clicking one of the icons in the small toolbar. The available functions depend on whether the fragment is a simple fragment or a fragment with an object reference. Object references refer to information from other objects which is integrated by means of special attributes of the data type “Object content”, “Object source”, or “OID”.

The following describes how to

- edit simple fragments, see the following section
- edit object references, see section “Editing Fragments with Object References” on page 302

Editing Simple Fragments

Changing the value of a fragment

To change the value of a fragment:

1. Start InSite Editing.
2. Navigate to the object with the fragment that you want to edit.
3. Place the mouse pointer on the toolbar of the fragment that you want to edit.

   The selected fragment is highlighted by a red frame.
4. Choose *Change value* on the context menu or click the icon .

   An excerpt from the Metadata dialog box opens. The input element's type depends on the data type (see table “Data types with formats used” on page 436).

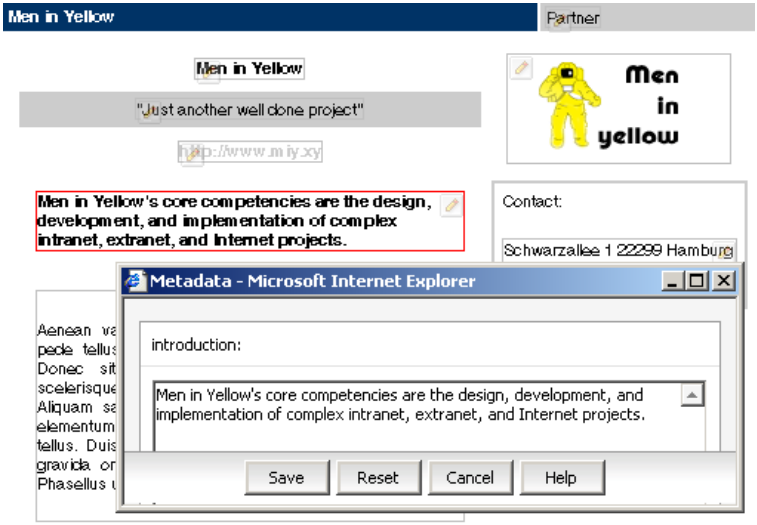

Men in Yellow
Partner
Men in Yellow
"Just another well done project"
Men in yellow
Men in Yellow's core competencies are the design, development, and implementation of complex intranet, extranet, and Internet projects.
Contact:
Schwarzallee 1 22299 Hamburg
Metadata - Microsoft Internet Explorer
introduction:
Men in Yellow's core competencies are the design, development, and implementation of complex intranet, extranet, and Internet projects.
Save
Reset
Cancel
Help

Fig. 87 – Editing an object fragment via InSite Editing

5. Make the desired changes.
6. Click the *Save* button.

Using the default value from the template

The default value is the value that a metadata item has in the template. For using the default value from the template, the `default` parameter must be set for the WCM tag used for integrating the fragment (see section “The `default` Parameter” on page 452).

To adopt the default value from the template:

1. Start InSite Editing.
2. Navigate to the object with the fragment that you want to edit.
3. Place the mouse pointer on the toolbar of the fragment that you want to edit.

   The selected fragment is highlighted by a red frame.

4. Choose *Use default value* on the context menu.
5. Confirm the security prompt by clicking the *Yes* button.

   The value of the metadata item is deleted. On the generated page, the value from the template is displayed.

Editing Fragments with Object References

For editing fragments containing object references, the context menu provides several functions. Additionally, icons offer access to the most important functions.

Replacing the referenced object with an existing object

To replace the referenced object with an existing object:

1. Start InSite Editing.
2. Navigate to the object with the fragment that you want to edit.
3. Place the mouse pointer on the toolbar of the fragment that you want to edit.

   The selected fragment is highlighted by a red frame.
4. Choose *Select different object* on the context menu or click the icon .

   A window for selecting an object opens.

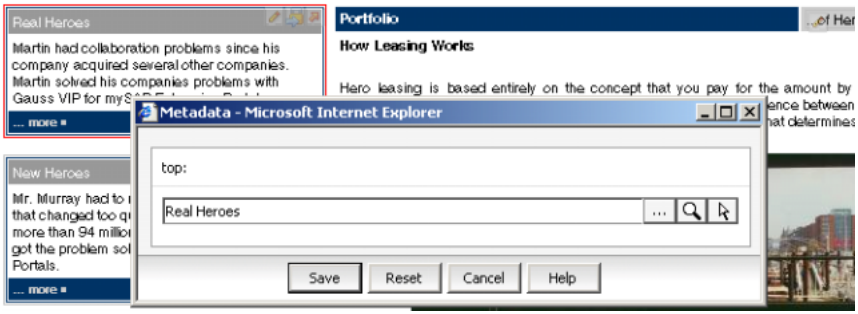


Fig. 88 – Replacing the referenced object via InSite Editing

5. Select the desired object.
6. Click the *OK* button.

Replacing the referenced object with a new object

You can also create a new object and replace the referenced object with this new object.

To replace the referenced object with a new object:

1. Start InSite Editing.
2. Navigate to the object with the fragment that you want to edit.
3. Place the mouse pointer on the toolbar of the fragment that you want to edit.

   The selected fragment is highlighted by a red frame.
4. Click the *assembled object* icon.
5. Follow the instructions of the wizard that guides you through the process. For detailed information, refer to the online help of InSite Editing.

   The new object is automatically entered in the fragment's reference.

Application Example: The HeroLeasing Website

The HeroLeasing website is a good example for demonstrating the functions of InSite Editing and the process of editing object fragments. The HeroLeasing website is supplied with Livelink WCM Server. It is located in the directory **{WCM installation directory}\import\HeroLeasing\archive** and can be imported into the WCM system. For further information on importing websites, refer to the Livelink WCM Server Administrator Manual.

Scenario

For the HeroLeasing website, the object types “Heropage” and “Herotemplate” were created. The new attribute set “Heropage” is assigned to both object types. The attribute set “Heropage” contains the attributes “top”, “middle”, and “bottom” of the data type “Object content”. For further information on adding object types and attribute sets, refer to chapter 16 “Managing Websites in the Content Client”.

On the basis of the object type “Herotemplate”, the template “Herotemplate” was created. On all pages of the website, three teasers for data sheets are to be displayed in the left window pane. This is realized by means of the attributes “top”, “middle”, and “bottom” integrated into the template by means of WCM tags. To enable InSite Editing for these attributes, the `edit` parameter is set in the WCM tags. For detailed information on using WCM tags, refer to chapter 15 “Working with WCM Tags”.

The teasers to be displayed on the website are saved as WCM objects of type “HTML page” in the topic “fragments” with the OID 52. When a teaser is replaced, only WCM objects from this topic should be available. For this reason, the `filter` parameter is also set in the WCM tags (see section “The `filter` Parameter” on page 451).

The parameter `default` ensures that the same teasers are displayed on all pages of the website (see section “The `default` Parameter” on page 452). The WCM tag for the attribute “top” looks as follows:

```
{VIPTOP edit="on" filter="(topic=52)" default="on"}.
```

In the “Content (WYSIWYG)” view of the Content client, the template “Herotemplate” looks like this:

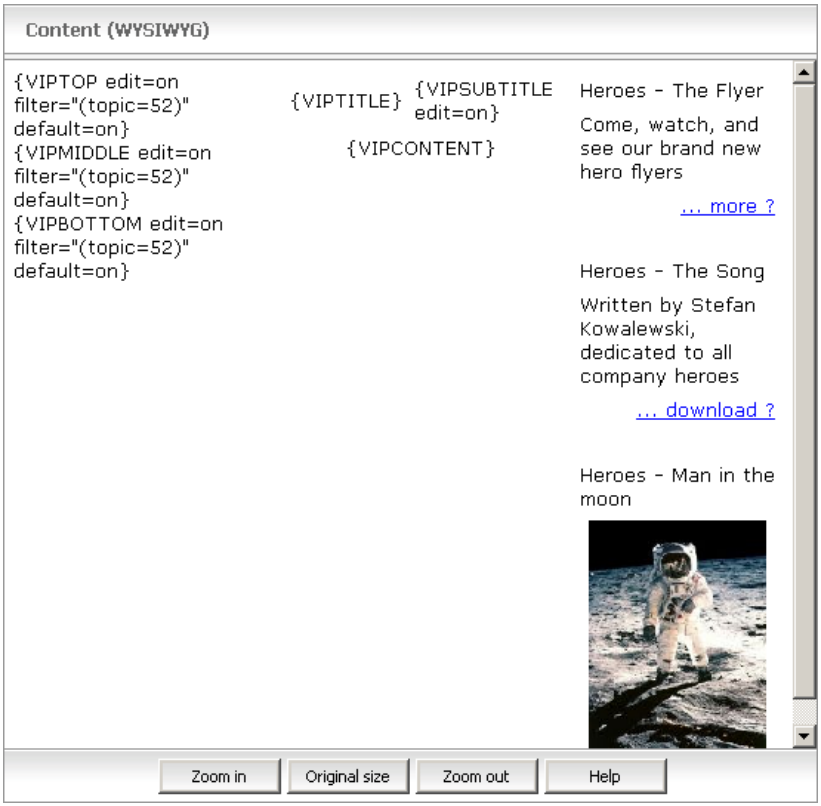


Fig. 89 – The template “Herotemplate” in the “Content (WYSIWYG)” view

Task: Edit Teasers

It is your task to check the teasers. You want to get a realistic impression of the objects and browse through the Edit view of the HeroLeasing website. You want to correct the errors you find directly in the website without starting the Content client. You should make your changes in the template “Herotemplate” to ensure that the same teasers are displayed on all pages of the website.

Procedure

1. Start InSite Editing.
2. Choose *Select object* on the context menu or click the corresponding icon.

   A window with the topic structure opens.
3. Select the template “Herotemplate”.

   In the left window pane, the three teasers are displayed.

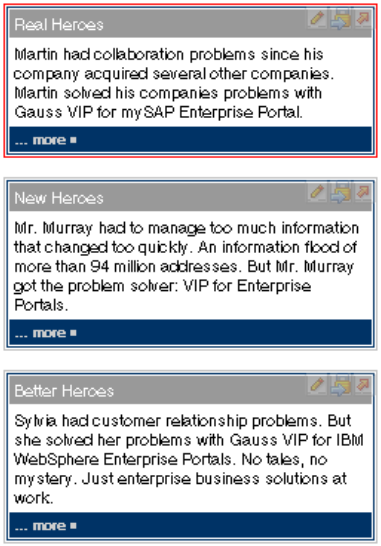

Real Heroes
Martin had collaboration problems since his company acquired several other companies. Martin solved his companies problems with Gauss VIP for mySAP Enterprise Portal.
... more
New Heroes
Mr. Murray had to manage too much information that changed too quickly. An information flood of more than 94 million addresses. But Mr. Murray got the problem solver: VIP for Enterprise Portals.
... more
Better Heroes
Sylvia had customer relationship problems. But she solved her problems with Gauss VIP for IBM WebSphere Enterprise Portals. No tales, no mystery. Just enterprise business solutions at work.
... more

Fig. 90 – Teasers on the website "HeroLeasing"

4. Place the mouse pointer on the fragment that you want to edit.

   The selected fragment is highlighted by a red frame.

5. Choose *Check out* on the context menu or click the icon .

   Depending on the configuration of the Content client, the referenced WCM object is opened in the integrated HTML editor or in the associated editor program.

6. Edit the object in the respective editor.

7. Check the object back in to the WCM system. Choose *Check in* on the context menu or click the icon .

8. To submit the changed object to Quality Assurance, choose *Submit* on the context menu.

After the object has been released by Quality Assurance, the changed teaser is available in the Production view on all pages of the website.

9. Repeat steps 4 to 6 for all other teasers that you want to change.

Task: Replace Teasers

It is your task to replace an old teaser with a new one. As all other teasers, the new teaser was saved as a WCM object of type “HTML page” in the topic “fragments”. You should make the changes directly in the template “Herotemplate” to ensure that the new teaser is displayed on all pages of the website.

Procedure

1. Start InSite Editing.
2. Choose *Select object* on the context menu or click the corresponding icon.

A window with the topic structure opens.

3. Select the template “Herotemplate”.

In the left window pane, the three teasers are displayed.

4. Place the mouse pointer on the toolbar of the fragment whose content you want to replace.

The selected fragment is highlighted by a red frame.

5. Choose *Select different object* on the context menu or click the icon .

A window showing all objects below the topic “fragments” opens.

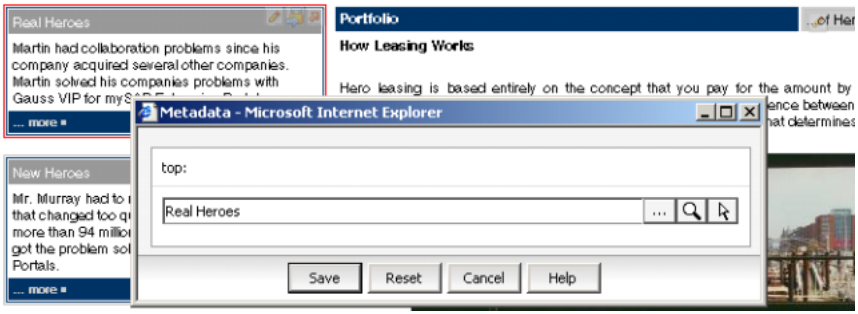


Fig. 91 – Replacing a teaser in the template “Herotemplate”

6. Select the desired object.
7. Click the *OK* button.
8. To submit the changed template “Herotemplate” to Quality Assurance, choose *Submit* on the context menu.

   After the template has been released by Quality Assurance, the new teaser is available in the Production view on all pages of the website.

Configuring InSite Editing

By means of an XML file, you can configure InSite Editing and thus customize it to suit your company's requirements. The following table provides a short overview of the different configuration items. A sample configuration is located in the directory **{WCM installation directory}\examples\insite-editing**.

Table 19 – Configuration items of InSite Editing

Item in the configuration file	Explanation
`<header display="true" type="default" oid="" />`	This item configures the header of InSite Editing. `display` attribute: - `true`: shows the header - `false`: hides the header `type` attribute: - `default`: The header has the Livelink layout. - `content`: The header shows the generated content of a referenced object; WCM tags are replaced by the metadata of the referenced object. - `include`: same as `content`, with the difference that WCM tags are replaced by the metadata of the current object `oid` attribute: OID of the object referenced for `content` or `include`
`<toolbar display="true" type="bottom" config="/resource/config/insite_toolbar.xml" />`	This item configures the toolbar of InSite Editing. `display` attribute: - `true`: shows the toolbar - `false`: hides the toolbar `type` attribute: - `none`: No text is shown on the icons. - `bottom`: The text is shown below the icon. - `right`: The text is shown to the right of the icon. `config` attribute: Path to the file in which you specified the icons to be displayed in the toolbar

Item in the configuration file	Explanation

`<menus>`
`  <menu name="main" display="true"`
`  config="/resource/config/deploymentmenu.xml" />`
`</menus>`
	This item configures the context menus of InSite Editing.
	
	`name` attribute:
	- `main`: context menu for normal WCM objects
	- `metadata`: context menu for simple object fragments
	- `object`: context menu for object fragments containing an object reference
	
	`display` attribute:
	- `true`: The context menu is available.
	- `false`: The context menu is not available.
	
	`config` attribute: Path to the file in which you specified the functions to be available on the context menu

`<summary display="false" width="200" height="200">`
`  <attribute name="title" />`
`</summary>`
	Use this item to determine whether the object information window is displayed for the current object.
	
	`display` attribute:
	- `true`: displays the object information window
	- `false`: hides the object information window
	
	`width` attribute: width of the window in pixels
	
	`height` attribute: height of the window in pixels
	
	`name` attribute: name of the metadata items to be displayed for an object

Item in the configuration file	Explanation
<buttons> <button name="new" display="true" action="CreateAssemblyMenu" url="/images/insite/set2/new.gif" label="HCL_BUTTON_NEW_FROM_CATEGORY"/> </buttons>	This item configures the little toolbar for object fragments. `name` attribute: - `new`: the *Assembled object* icon - `edit`: the *Change value* or *Select different object* icon - `checkin`: the *Check out* icon - `checkout`: the *Check in* icon - `goto`: the *Go to object* icon - `param`: the *WCM tags* icon `display` attribute: - `true`: The icon is displayed. - `false`: The icon is hidden. To display the icon for specific object fragments again, configure the `buttons` parameter for the WCM tag (see section “The `buttons` Parameter” on page 460). `url` attribute: path to the icon's image file `label` attribute: The user interface texts of InSite Editing are stored as key-value pairs in the so-called properties files. This attribute is the key of the icon text, the associated value is the translated icon text. For detailed information on translating user interface texts, refer to the Livelink WCM Server Administrator Manual (chapter “Working in the Admin Client”).

To configure InSite Editing:

1. Create a configuration file with the desired items. You have a free choice of name for the file.
2. Copy the configuration file to the directory **\vipimages** of the InSite Editing deployment system.

3. In the Admin client, configure the `VipHCLApplication`.

   For this purpose, add a new parameter node with the name of the InSite Editing deployment system. Add a `config` parameter to this node. As the parameter value, enter `{name of configuration file}`.

   For detailed information on configuring applications, refer to the online help of the Admin client.

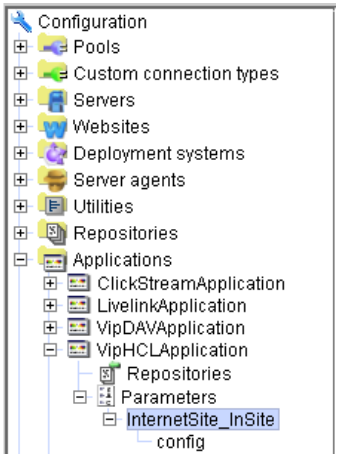


Fig. 92 – Configuring the `VipHCLApplication`

4. Shut down the JSP engine.
5. Delete all generated files from the directory **\200** of the InSite Editing deployment system.
6. Restart the JSP engine.

Switching off InSite Editing for individual pages

The scripts generated into the page for InSite Editing may collide with elements of cascading style sheets. This may result in the page not being displayed correctly. In this case, you should deactivate InSite Editing for the respective page. For this purpose, specify the tag `{VIPINSITEEDITINGOFF}` in one of the templates of the WCM object.

CHAPTER 10
Working with Assembled Objects

Quite often, the layout and content of web pages are not freely designable, but are subject to a clearly defined structure that must absolutely be adhered to. An example are press releases which do not only always have the same layout, but also consist of identical parts, such as title, introduction, main text, date, and image. You can define accurate specifications for such objects in the WCM system. This enables authors to easily compose web pages.

Defining object structures as well as adding and editing objects based on these object structures is called *Content Assembly*. The object type “Assembled object” was introduced to represent object structures that may consist of the content of several objects and additional properties.

An object of type “Assembled object” does not have any content and thus cannot be checked out. All data belonging to the object is saved in the metadata and is displayed in the generated page by means of WCM tags integrated in the template. For this reason, an object of type “Assembled object” must always be assigned a template.

The functions of Content Assembly can be used in both the Content client and InSite Editing. In this chapter, you get information on the following topics:

- how to work with Content Assembly, see the following section
- editing assembled objects, see section “Editing Assembled Objects” on page 324

- two examples of using the Content Assembly functions, see section “Application Examples” on page 324

> **Note:** The functions of Content Assembly are not limited to assembled objects. They can also be used in connection with other objects. For an application example, refer to section “Application Example: The HeroLeasing Website” starting on page 304.

General Procedure

Before you can use Content Assembly, you must perform the following steps.

Step 1 – Add an object category (responsible: web designer)

First you determine which information the assembled object will be composed of. For this purpose, you create an object category. Each information of the assembled object is represented by a property of this object category. If you want to use Content Assembly for creating press releases, for example, you require an object category named “PressRelease”, which contains the properties “Title”, “Introduction”, “MainText”, “Date”, and “Image”.

The property's data type controls the format of the information on the generated page. The following table lists the most important data types and the default formats used by Livelink WCM Server.

Table 20 – Important data types with formats used

Data type	Format used
Date	Returns – depending on the language setting of the WCM object – the date in the following form: 04.09.2003 (German) Sep 4, 2003 (English)
External URL	Returns the URL itself
Formatted text	Returns the HTML content of the attribute. Only the content of the body element is returned. If there is no body element, the entire text is generated to the page. Links contained in the text are not managed automatically.
Integer	Returns the integer itself The value range is defined as follows: –2147483647 <= integer <= 2147483647
List	Returns a comma-separated list of the entries in the order they were entered. In the case of expressions enclosed in quotes, quotes are not generated to the page.
Livelink ID	Returns the ID of the object in the Livelink system
Set	Returns an alphabetically sorted, comma-separated list of the entries. Double entries are removed. In the case of expressions enclosed in quotes, quotes are not generated to the page.
Set of e-mail addresses	Returns an alphabetically sorted, comma-separated list of the entries. Double entries are removed. In the case of expressions enclosed in quotes, quotes are not generated to the page.

Data type	Format used
Object content	Returns the generated content of the referenced object. Only the content of the body element is returned. WCM tags are replaced by the metadata of the **referenced** object.
Object source	Returns the generated content of the referenced object. Only the content of the body element is returned. WCM tags are replaced by the metadata of the **current** object.
Note: When integrating WCM objects by means of special attributes of type “Object content” or “Object source”, it may happen that a user sees the content of an object without having read rights for the object. Example: The content of object A is integrated in object B by means of a WCM tag. The user only has read rights for object B. However, via the respective page, the user can also see the content of object A.	
Object URL	Returns – depending on the settings of the deployment system – the URL of the referenced object as a relative, absolute, or server-relative reference.
OID	Returns the OID as a string
Text	Returns the text itself
Time	Returns – depending on the language setting of the WCM object – the time in the following form: 15:59:23 (German) 3:59:23 PM (English)
Boolean value	Returns – depending on the language setting of the WCM object – `true` or `false` or `wahr` or `falsch`.

Data type	Format used
String	Returns the entry itself
Time stamp	Returns – depending on the language setting of the WCM object – the date and time in the following form: 04.09.2003 15:59:23 (German) Sep 4, 2003 3:59:23 PM (English)

The properties of the sample object category “PressRelease” could have the following data types:

Table 21 – Properties of the object category “PressRelease”

Property	Data type
TitleArticle	String
Introduction	Text
MainText	Formatted text
Image	Object URL
Date	Date

For detailed information on adding object categories and properties, refer to section “Managing Object Categories” on page 485.

Step 2 – Create a template (responsible: web designer)

The next step is to define the layout for the assembled object. Create a template and assign the object category created in step 1 to this template. Use WCM tags to integrate the object category's properties (and thus the information of the assembled object) in the template. Arrange the WCM tags in the template as you want them to be displayed in the generated page.

The following figure illustrates the structure of the press release template.

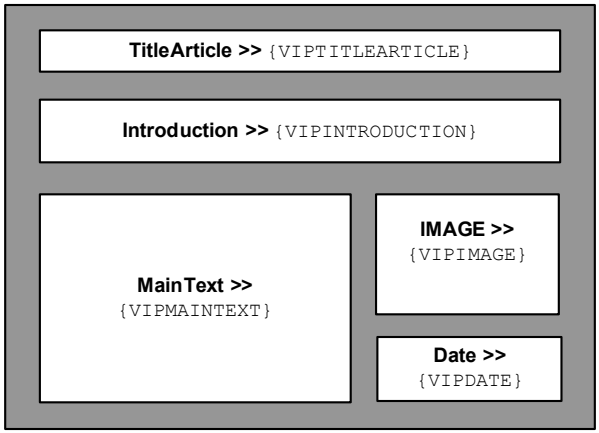


Fig. 93 – Structure of the "PressRelease" template

By means of the parameters for the WCM tags, you can activate InSite Editing for the individual metadata items and specify default values. For detailed information on WCM tags, refer to chapter 15 "Working with WCM Tags".

Step 3 – Add an assembled object (responsible: editor)

To add assembled objects based on the category created in step 1, choose *Object* → *New* → *assembled object*. Follow the instructions of the wizard. For detailed information, refer to the online help of the Content client.

The new objects have the properties of the assigned object category. To edit these properties – and thus the content of the assembled object – go to the *Properties* tab of the Metadata dialog box. There you can, for example, specify the OID of the object which is to be used as the image for the press release.

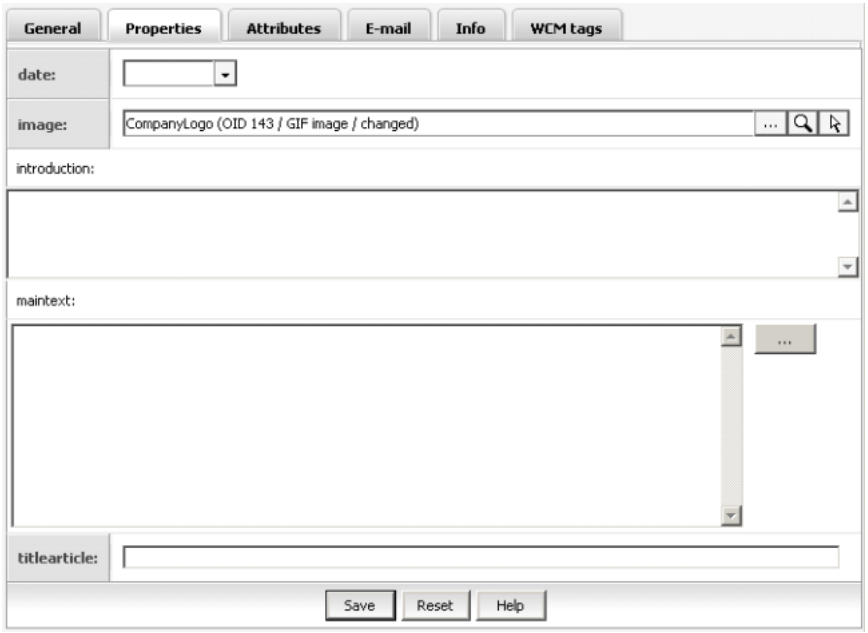


Fig. 94 – The *Properties* tab in the Metadata dialog box

Editing Assembled Objects

In contrast to “normal” WCM objects, assembled objects do not have a content component. Thus, the following functions are not available for assembled objects:

- Check out
- Check in
- Edit content
- Undo
- Replace
- Add reference

All information of the assembled object is managed in the Metadata dialog box and can be edited as usual.

Application Examples

For a better understanding, the following two application examples show how to work with assembled objects.

> **Note:** The examples described below are based on the example website “HeroLeasing”, which is supplied with Livelink WCM Server and can be imported into the WCM system. The website is located in the directory **{WCM installation directory}\import\HeroLeasing\archive**. For further information on importing websites, refer to the Livelink WCM Server Administrator Manual.

Content Assembly with the Content Client

This example describes a typical scenario for using the functions of Content Assembly and demonstrates how to proceed when adding assembled objects via the Content client.

Scenario

For realizing special tasks and on-site projects, your company cooperates with several partner companies. On the website of your company, these partners are presented with their areas of expertise, experiences, and contact data. The layout of these partner profiles should be consistent. For each partner, the following information should be displayed:

- name
- slogan
- short introduction
- URL of the website
- detailed description
- contact person and address
- logo

Prerequisites

- You have the access rights “Read” and “Create” for the topics to which you want to add the template for the partner objects and the partner objects themselves.
- You are assigned the functional areas “Dynamic” and “Advanced” (or the functional areas your WCM administrator has provided for adding templates and assembled objects). To check your functional areas, choose *Tools* → *Login info*.
- You have the administration right “Website administration”.

Procedure

Step 1 – Create the object category “Partner”

Add the object category “Partner” with the respective properties. The following table shows the properties used in this example with their data types and the corresponding WCM tags.

Table 22 – Properties of the object category “Partner”

Property	Data type	WCM tag
partnername	String	{VIPPARTNERNAME}
slogan	String	{VIPSLOGAN}
introduction	Text	{VIPINTRODUCTION}
partnerurl	External URL	{VIPPARTNERURL}
partnerdescription	Formatted text	{VIPPARTNERDESCRIPTION}
contact	Text	{VIPCONTACT}
logo	Object URL	{VIPLOGO}

Step 2 – Create the template

Define the layout for the partner profiles by means of a template. For this purpose, add a template of type “HTML template” and assign the object category ”Partner“ to this template. In our example, this template has the title “Partnertemplate”.

Check out the template and edit it according to your guidelines. Integrate the properties defined in the first step into the template by means of WCM tags.

In principle, the body section of the template “Partnertemplate” looks as follows:

```
<body>

<table width="100%" border="0" cellspacing="2" cellpadding="3">
  <tr>
    <td><div align="center"><center><table border="0" cellspacing="3"
     cellpadding="3">
      <tr>
        <td align="center" class="frame-dark"><b>{VIPPARTNERNAME
         edit=on default=on}</b></td>
        <td rowspan="3" valign="middle" align="center"
         class="frame-dark">
        {VIPLOGO edit=begin filter="(topic=8)" thumbnails=on}
         <img src="{VIPLOGO default=on}" border="0" width="150"
         height="73">{VIPLOGO edit=end} </td>
      </tr>
      <tr>
        <td align="center" class="lightgrey">{VIPSLOGAN edit=on
         default=on}</td>
      </tr>
      <tr>
        <td align="center" class="rahmen">{VIPPARTNERURL edit=begin
         default=on}<a href="{VIPPARTNERURL}"target="_blank">
         {VIPPARTNERURL}</a>{VIPPARTNERURL edit=end default=on}</td>
      </tr>
      <tr>
        <td class="frame" valign="top"><table border="0" cellpadding="8"
         cellspacing="0">
         <tbody>
          <tr>
            <td valign="top"><b>{VIPINTRODUCTION edit=on
            default=on}<br><br></b>
            {VIPPARTNERDESCRIPTION edit=on default=on}</td>
          </tr>
         </tbody>
        </table>
        </td>
        <td valign="top"><table width="100%" border="0" cellspacing="1"
         cellpadding="1">
          <tr>
            <td class="lightgrey" valign="top"><table width="100%"
            border="0" cellspacing="1" cellpadding="5">
              <tr>
                <td bgcolor="#FFFFFF">Contact:<p>
                {VIPCONTACT edit=on default=on}</td>
              </tr>
            </table>
            </td>
```

```
          </tr>
        </table>
        <p></td>
      </tr>
    </table>
    </center></div></td>
  </tr>
</table>
</body>
```

No {VIPCONTENT} tag is required for the template. As the entire data of the partner is contained in the properties from the object category, there is no explicit content.

Steps 3 – Determine default values

You have the possibility to define default values for the individual properties in the template. In this way you can point the editor responsible for the partner object to the information the editor is to enter in the individual fields. The default values from the template are adopted by the objects based on the template if the respective field of these objects is empty. The default values are not shown in the Metadata dialog box of the objects; they are only used for generating the corresponding pages.

To determine default values:

1. Open the “Partnertemplate” template in the integrated HTML editor.
2. In the *Design* mode, select a WCM tag, such as {VIPPARTNERNAME}, and choose *Edit WCM tag* on the context menu.

   The wizard for editing WCM tags starts.
3. For *Enable InSite Editing*, click the *on* radio button.
4. For *Default value*, click the *on* radio button.
5. Repeat steps 2 to 4 for all WCM tags for which you want to set default values.

6. Enter the desired default values on the *Properties* tab of the Metadata dialog box.

In the object preview, the template with the default values looks as follows:

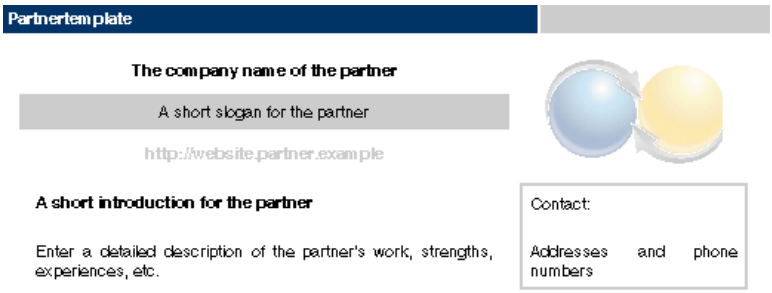

Partnertemplate
The company name of the partner
A short slogan for the partner
http://website.partner.example
A short introduction for the partner
Enter a detailed description of the partner's work, strengths, experiences, etc.
Contact:
Addresses and phone numbers

Fig. 95 – The template “Partnertemplate” in the object preview

Step 4 – Add partner objects

On the basis of the “Partnertemplate” template, add WCM objects of type “Assembled object”. Enter the respective information on the partner as values for the properties. The following figure shows the properties of the partner “Kryptonite Power GmbH” as an example.

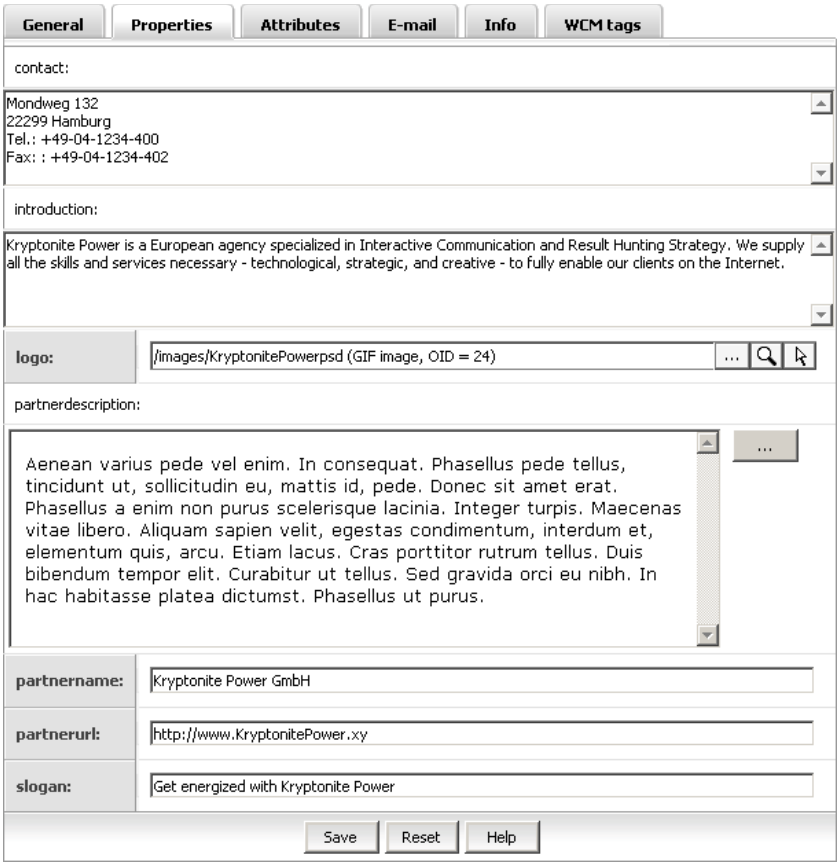

General
Properties
Attributes
E-mail
Info
WCM tags
contact:
Mondweg 132
22299 Hamburg
Tel.: +49-04-1234-400
Fax: : +49-04-1234-402
introduction:
Kryptonite Power is a European agency specialized in Interactive Communication and Result Hunting Strategy. We supply all the skills and services necessary - technological, strategic, and creative - to fully enable our clients on the Internet.
logo:
/images/KryptonitePowerpsd (GIF image, OID = 24)
partnerdescription:
Aenean varius pede vel enim. In consequat. Phasellus pede tellus, tincidunt ut, sollicitudin eu, mattis id, pede. Donec sit amet erat. Phasellus a enim non purus scelerisque lacinia. Integer turpis. Maecenas vitae libero. Aliquam sapien velit, egestas condimentum, interdum et, elementum quis, arcu. Etiam lacus. Cras porttitor rutrum tellus. Duis bibendum tempor elit. Curabitur ut tellus. Sed gravida orci eu nibh. In hac habitasse platea dictumst. Phasellus ut purus.
partnername:
Kryptonite Power GmbH
partnerurl:
http://www.KryptonitePower.xy
slogan:
Get energized with Kryptonite Power
Save
Reset
Help

Fig. 96 – Entering partner information for the assembled object during creation

On the HTML page of the partner object, the information entered is now displayed as defined in the template. In the object preview, the profile of the partner “Kryptonite Power GmbH” looks as follows:

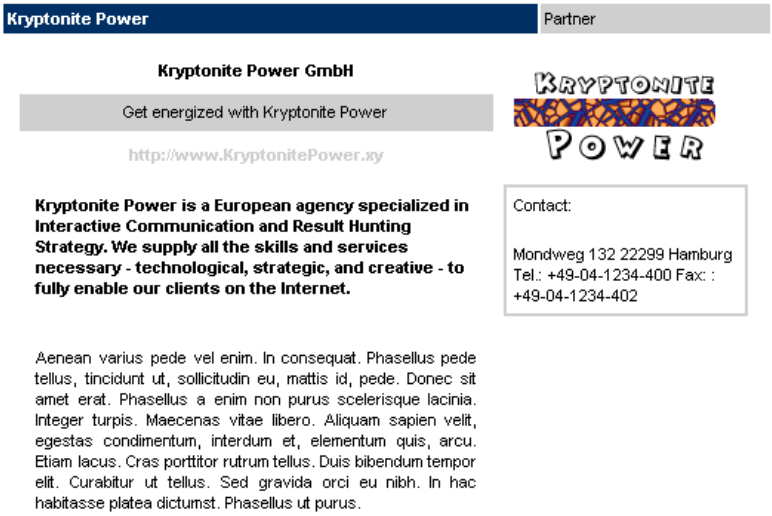

Kryptonite Power
Partner
Kryptonite Power GmbH
Get energized with Kryptonite Power
http://www.KryptonitePower.xy
KRYPTONITE
POWER
Kryptonite Power is a European agency specialized in Interactive Communication and Result Hunting Strategy. We supply all the skills and services necessary - technological, strategic, and creative - to fully enable our clients on the Internet.
Contact:
Mondweg 132 22299 Hamburg
Tel.: +49-04-1234-400 Fax: : +49-04-1234-402
Aenean varius pede vel enim. In consequat. Phasellus pede tellus, tincidunt ut, sollicitudin eu, mattis id, pede. Donec sit amet erat. Phasellus a enim non purus scelerisque lacinia. Integer turpis. Maecenas vitae libero. Aliquam sapien velit, egestas condimentum, interdum et, elementum quis, arcu. Etiam lacus. Cras porttitor rutrum tellus. Duis bibendum tempor elit. Curabitur ut tellus. Sed gravida orci eu nibh. In hac habitasse platea dictumst. Phasellus ut purus.

Fig. 97 – Partner profile in the object preview

You can modify the partner profiles on the *Properties* tab of the Metadata dialog box at any time.

Content Assembly with InSite Editing

This example describes how to use the functions of InSite Editing in order to edit assembled objects directly in the website.

Scenario

Your company has several partner companies, which are introduced on the website. You want to edit these partner profiles directly in the website without starting the Content client. When changing the logo, a selection list of all available logos located below a certain topic should be displayed.

Prerequisites

- InSite Editing is configured for your website. For more information, refer to chapter 9 “InSite Editing”.
- You have the access rights “Read” and “Change metadata” for the partner objects you want to edit.
- You have the access rights “Read”, “Change object”, and “Change metadata” for the template the partner objects are based on.
- You are assigned the functional areas “Dynamic” and “Advanced” (or the functional areas your WCM administrator has provided for editing templates and assembled objects). To check your functional areas, choose *Tools* → *Login info*.

Procedure

Step 1 – Modify partner template

First, you must enable InSite Editing for the individual fields of the partner objects. Proceed as follows:

1. Open the “Partnertemplate” template in the integrated HTML editor.
2. In the *Design* mode, select a WCM tag and choose *Edit WCM tag* on the context menu.

   The wizard for editing WCM tags starts.
3. For *Enable InSite Editing*, click the *on* radio button.
4. Repeat steps 2 and 3 for all desired WCM tags.

Step 2 – Configure selection and thumbnails for logo

When a partner logo is changed, the entire topic structure opens by default. You can restrict the selection by defining a filter in the WCM tag `{VIPLOGO}`. For example, the statement `filter="((topic=4).and.(oty_name=GIF))"` only shows objects of type “GIF image” below the topic with the OID 4.

> **Notes:**
>
> The filter expression may already contain double quotes (example: `oty_name="GIF"`). In this case, enclose the value of the `filter` parameter in single quotes (example: `filter='(oty_name="GIF")'`).
>
> For further information on creating filters, refer to section “The Filter Editor” on page 376. For notes on using the `filter` parameter, refer to section “The `filter` Parameter” on page 451.

To configure a thumbnail view for the logos, use the `thumbnails` parameter. If the `thumbnails` parameter has the value `on`, thumbnails are automatically displayed for graphic objects.

The WCM tag for integrating the logo looks as follows:

```
{VIPLOGO edit=begin filter="((topic=4).and.(oty_name=GIF))"
  thumbnails=on default=on}
  <img src="{VIPLOGO}" align="middle" />
{VIPLOGO edit=end}
```

Step 3 – Start InSite Editing

Start InSite Editing. See section “Starting InSite Editing” on page 295.

Step 4 – Edit partner object

1. Go to the partner object that you want to modify (here: the profile of the partner “Kryptonite Power GmbH”).

   All fragments of the partner object that you may change and for which InSite Editing is enabled are highlighted by a red frame.

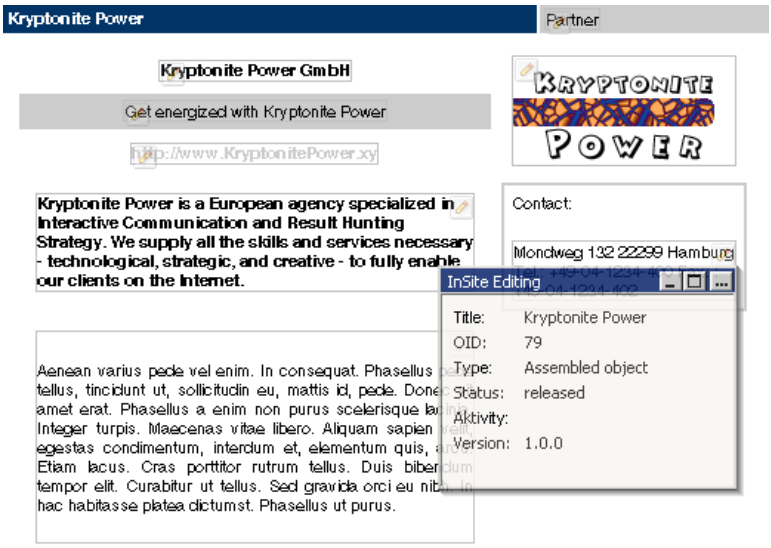

Kryptonite Power
Partner
Kryptonite Power GmbH
Get energized with Kryptonite Power
http://www.KryptonitePower.xy
KRYPTONITE POWER
Kryptonite Power is a European agency specialized in Interactive Communication and Result Hunting Strategy. We supply all the skills and services necessary - technological, strategic, and creative - to fully enable our clients on the Internet.
Contact:
Monckweg 132 22299 Hamburg
InSite Editing
Title: Kryptonite Power
OID: 79
Type: Assembled object
Status: released
Aktivity:
Version: 1.0.0
Aenean varius pede vel enim. In consequat. Phasellus tellus, tincidunt ut, sollicitudin eu, mattis id, pede. Done amet erat. Phasellus a enim non purus scelerisque la Integer turpis. Maecenas vitae libero. Aliquam sapien egestas condimentum, interdum et, elementum quis, Etiam lacus. Cras porttitor rutrum tellus. Duis biben tempor elit. Curabitur ut tellus. Sed gravida orci eu nib hac habitasse platea dictumst. Phasellus ut purus.

Fig. 98 – Partner object in InSite Editing

2. Click the icon next to the fragment that you want to edit.

   A dialog box in which you can change the value of the fragment opens.

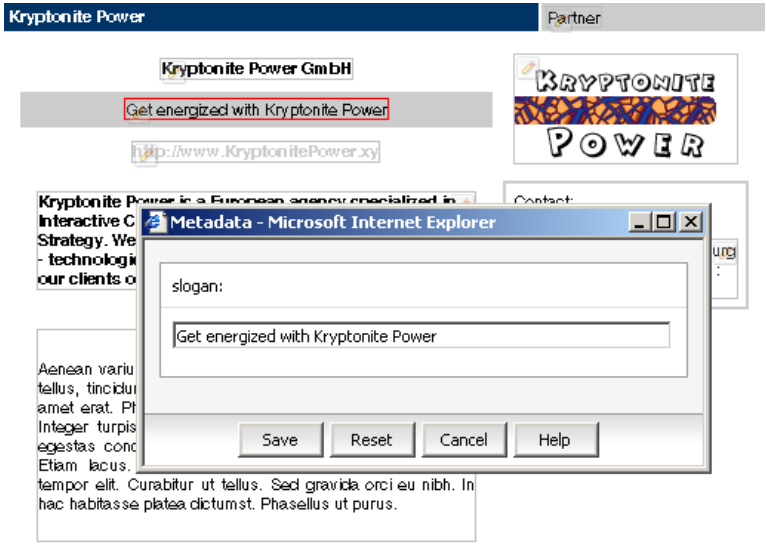


Fig. 99 – Editing a partner object via InSite Editing

3. Make the desired changes.
4. Click the *Save* button.

   The changes immediately become available in the Edit view of the website.
5. To make further changes, repeat steps 2 to 4.

Step 5 – Replace logo

1. Next to the logo, click the icon .

   A dialog box in which you can replace the logo opens.
2. Enter the OID of the new logo. If you do not know the OID, click the icon .

   A dialog box with a list of all available logos opens.

Fig. 100 – Changing the partner logo via InSite Editing

3. Select the desired logo and click the *OK* button.

   The dialog box closes and the name of the selected logo is written to the field *logo*.

4. Click the *Save* button.

   The changes immediately become available in the Edit view of the website.

CHAPTER 11

Livelink Contents in the Content Client

Due to the integration of the WCM system with Livelink, you can add Livelink objects as WCM objects to a WCM-managed website. These WCM objects are called *relators*. They always refer to a certain version of the Livelink object. The WCM object type “Livelink relator” enables you to integrate a single Livelink object in a website; the WCM object type “Livelink folder relator” is provided for integrating Livelink folders.

You can directly access relators in the WCM system (e.g. with the Content client). The actual objects, however, are stored in the Livelink database. Relators are treated like normal WCM objects and are subject to staging. The reference management of Livelink WCM Server is also available for relators. Livelink folder relators have a special feature: As they do not have any content, you cannot check out and check in Livelink folder relators.

Livelink documents integrated in a website by means of relators are displayed directly in the browser. If files cannot be displayed directly in the browser, e.g. PDF files or Word documents, a so-called surrogate page containing a link to the file is generated. Livelink folders become WCM topics. As WCM topics have content, an HTML page is generated for each Livelink folder.

In the Navigation area of the Content client, the following icons symbolize relators:

The letters “LL” in the bottom right corner of the object type icon stand for Livelink. (The figure shows the relator icon for an Excel document).

If the system cannot access the Livelink object referenced by the relator, the relator's icon is marked by a red cross. The reason for this may be, for example, that the Livelink object has been deleted or that the connection between the WCM system and the Livelink system has been interrupted.

In this chapter, the following topics are introduced:

- adding relators, see the following section
- editing relators, see section “Editing Relators” on page 353
- deleting relators, see section “Deleting a Relator” on page 362

Adding Relators

There are different ways of adding Livelink objects as WCM objects. You can:

- add a single Livelink document as a WCM object, see the following section
- add an existing Livelink folder structure as WCM objects, see section “Adding a Livelink Folder Relator” on page 346
- add a Livelink document together with the associated relator, see section “Adding a Relator Together with the Associated Livelink Document” on page 350

Notes:

You cannot use the Content client to add folders in Livelink.

You should not **manually** add relators and folder relators below a folder relator. The process that synchronizes the WCM system with the Livelink system checks whether all the subfolders and documents in the Livelink folder are represented by relators; manually created relators are deleted.

Prerequisites for adding relators

- In the WCM system, you have the access rights “Read” and “Create” for the topic to which the relator is to be added.

  If you want to add a new Livelink document together with the relator, you also need the permission “Add Items” for the Livelink folder to which you want to add the object.

- You are assigned the functional area “Livelink”. To check your functional areas, choose *Tools* → *Login info*.

- Your WCM administrator prepared the website, in which you want to add the relator, for using Livelink functionality. For an automatic synchronization of WCM objects to Livelink objects, the WCM Kernel Integration module must be installed. For more information, refer to the Livelink Integration Manual.

Adding a Livelink Relator

Objective

You want to integrate a single Livelink object in a WCM-managed website. A wizard guides you through the process of adding the relator.

Starting the wizard

1. Change to the Edit view of the website.
2. In the Navigation area, mark the topic to which you want to add the relator.
3. Choose *Object* → *New* → *Relator*.

Specifying the Livelink system

There may be more than one Livelink system in your company. In the first dialog box, you specify the Livelink system in which the Livelink object that you want to integrate is stored.

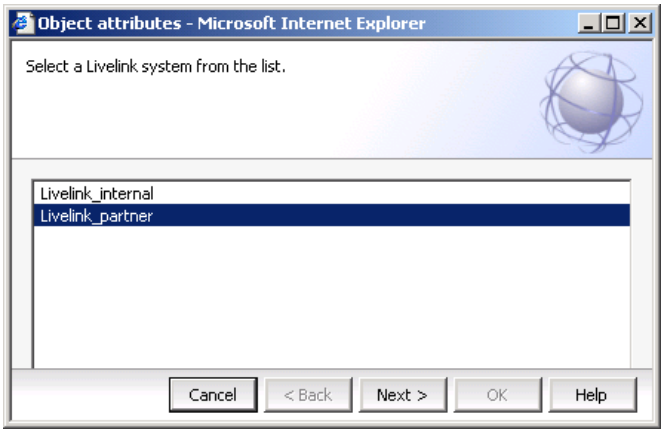


Fig. 101 – Specifying the Livelink system

Confirm your entries by clicking the *Next* button.

Selecting a Livelink object

In this dialog box, you select the Livelink object that is to be added as a relator in the WCM system.

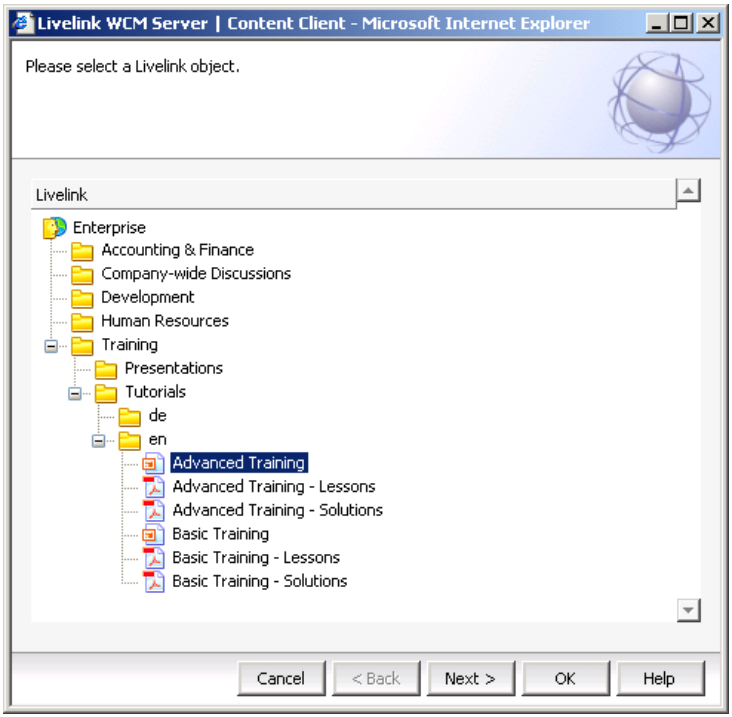


Fig. 102 – Selecting the Livelink object when adding a relator

Confirm your selection by clicking the *Next* button.

Specifying the version of the Livelink object

The Livelink object that the relator refers to may be subject to change. In this dialog box, you determine how the relator reacts when the Livelink object changes.

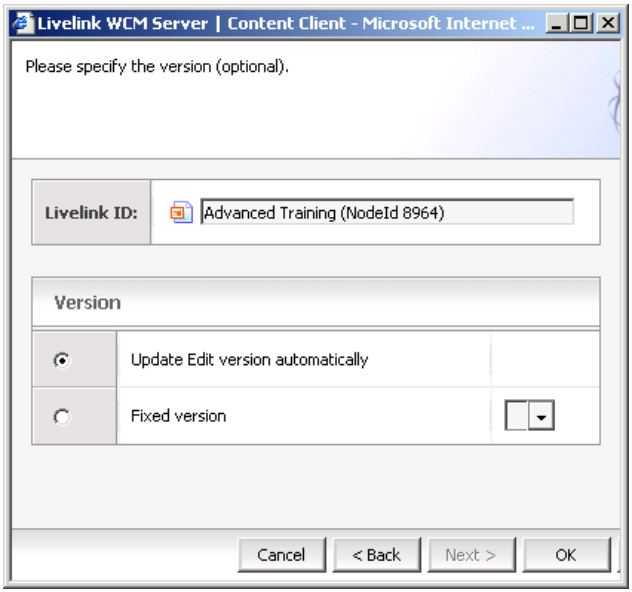


Fig. 103 – Specifying the version of the Livelink object

- *Livelink ID:* Livelink ID of the selected object. The Livelink ID is composed of the name and the ID of the object in the Livelink system. You cannot change this entry.
- *Update Edit version automatically:* If you click this radio button, the Edit view of the relator will be automatically changed whenever the object is modified in the Livelink system. For the changes to become visible in the published website, you must submit the changed relator which then must be released by Quality Assurance.

> **Note:** A relator can only be updated automatically if its status is either “changed”, “rejected”, “released”, or “delayed release”. If this requirement is not met, use the appropriate staging action to change the relator's status. After this, you can adapt the relator to the modified Livelink object by choosing *Object* → *Synchronize*.

- *Fixed version:* If you click this radio button, the relator will be based on a certain version of the Livelink object. When the object changes in the Livelink system, the relator remains unchanged. You can, however, update the relator manually via the Livelink metadata dialog box (see section “Modifying Livelink Metadata” on page 355).
- *Version:* If you clicked the *Fixed version* radio button, specify the desired version of the Livelink object by clicking it in the drop-down list.

Confirm your entries by clicking the *OK* button.

Entering a remark

In this dialog box, you may enter information for the relator's log.

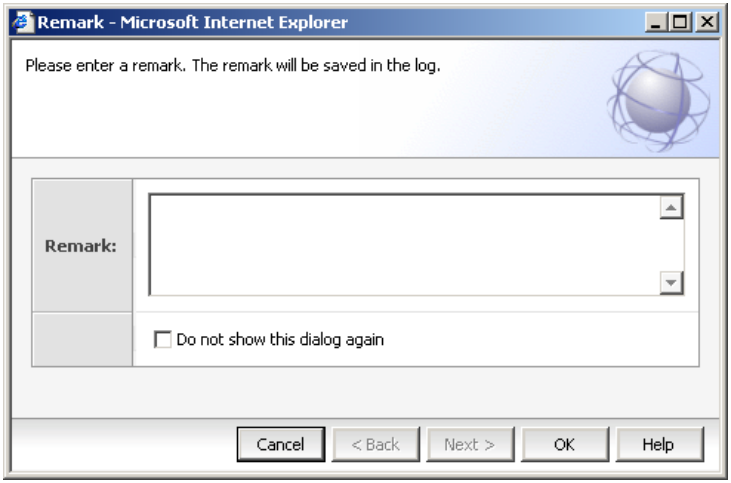


Fig. 104 – Entering comments about the relator

Confirm your entries by clicking the *OK* button.

The WCM system creates the relator and shows it in the Navigation area. The relator's status is “changed”.

For the relator to become visible in the published website, you must submit it. After this, the relator must be released by Quality Assurance.

Adding a Livelink Folder Relator

Objective

You want to integrate a Livelink folder including all subfolders and subordinate documents into the WCM system. Each Livelink object is to be represented by a relator in the WCM system. A wizard guides you through this task.

Starting the wizard

1. Change to the Edit view of the website.
2. In the Navigation area, mark the topic in which you want to integrate the Livelink objects.
3. Choose *Object* → *New* → *Relator*.

Specifying the Livelink system

There may be more than one Livelink system in your company. In the first dialog box, you specify the Livelink system in which the Livelink folder that you want to integrate is stored.

Selecting the Livelink folder

In this dialog box, you select a Livelink folder. The subfolders and documents contained in this folder will be integrated in the WCM system.

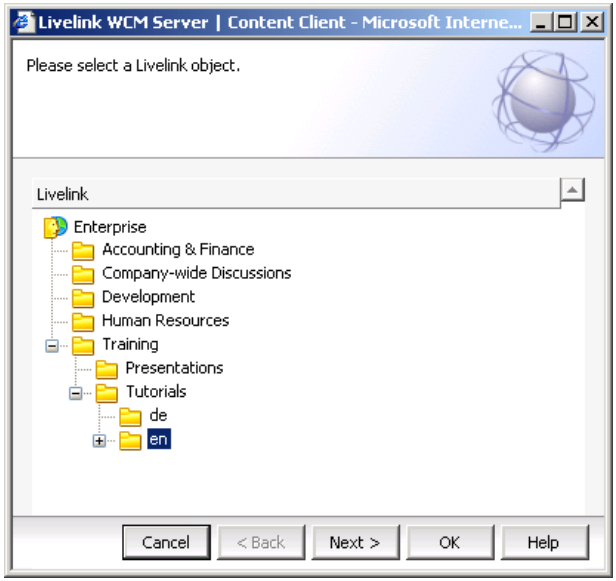


Fig. 105 – Selecting the Livelink folder

Confirm your selection by clicking the *Next* button.

Specifying the version of the Livelink folder

The Livelink folder that the Livelink folder relator refers to may be subject to change. In this dialog box, you determine how the Livelink folder relator reacts when the Livelink folder changes.

> **Note:** For each object in the Livelink folder, a separate relator is created. The setting that you make here, initially applies to all relators. However, you can subsequently change this setting for individual relators (see section “Modifying Livelink Metadata” on page 355).

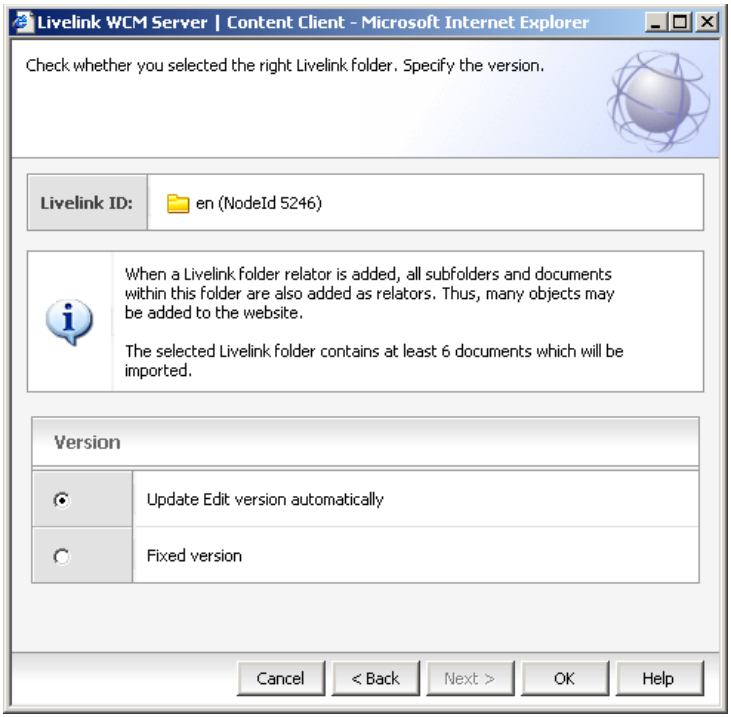

Livelink WCM Server | Content Client - Microsoft Internet Explorer
Check whether you selected the right Livelink folder. Specify the version.
Livelink ID:
en (NodeId 5246)
When a Livelink folder relator is added, all subfolders and documents within this folder are also added as relators. Thus, many objects may be added to the website.
The selected Livelink folder contains at least 6 documents which will be imported.
Version
Update Edit version automatically
Fixed version
Cancel
< Back
Next >
OK
Help

Fig. 106 – Specifying the version of the Livelink folder relator

- *Livelink ID:* Livelink ID of the selected folder. The Livelink ID is composed of the name and the ID of the folder in the Livelink system. You cannot change this entry.
- *Update Edit version automatically:* If you click this radio button, the Edit view of the Livelink folder relator will be automatically adapted whenever the metadata of the associated Livelink folder change. If an object is added to the Livelink folder, a new relator representing this object is automatically created. If an object is removed from the Livelink folder, the associated relator will be deleted.

  For the changes to become visible in the published website, you must submit the changed relator which then must be released by Quality Assurance.

  Note: A relator can only be updated automatically if its status is either “changed”, “rejected”, “released”, or “delayed release”. If this requirement is not met, use the appropriate staging action to change the relator's status. After this, you can adapt the relator to the modified Livelink object by choosing *Object* → *Synchronize*.

- *Fixed version:* If you click this radio button, the Livelink folder relator will be based on the current version of the Livelink folder. When the folder's metadata change in the Livelink system, the associated Livelink folder relator remains unchanged. If an object is added to the Livelink folder, a new relator representing this object is not automatically created. However, the relator for an object will be deleted, if the object is removed from the Livelink folder.

  You can adapt the relator manually via the Livelink metadata dialog box (see section“Modifying Livelink Metadata” on page 355).

Confirm your entries by clicking the *OK* button.

Entering a remark

In the following dialog box, you may enter information for the logs of the individual relators.

Confirm your entries by clicking the *OK* button.

The WCM system creates the relators and shows them in the Navigation area. Their status is “changed”.

For the relators to become visible in the published website, you must submit them. After this, the relators must be released by Quality Assurance.

Adding a Relator Together with the Associated Livelink Document

Objective

You want to integrate a relator in the WCM system which refers to a file which does not yet exist as a document in the Livelink system. Together with the relator, you want to add this file as a new document in Livelink. A wizard guides you through this process.

> **Notes:**
>
> Livelink objects that you add in the Content client together with the relators will **always** be deleted when the associated relator is destroyed.
>
> You cannot use the Content client to add folders in Livelink.

Starting the wizard

1. Change to the Edit view of the website.
2. In the Navigation area, mark the topic to which you want to add the relator.

3. Choose *Object* → *New* → *from object type* or click the appropriate icon.

Icon for adding an object on the basis of the object type

Specifying title, file, and object type

In the first dialog box, you specify the file which is to be added to the Livelink system and the name of the new Livelink document.

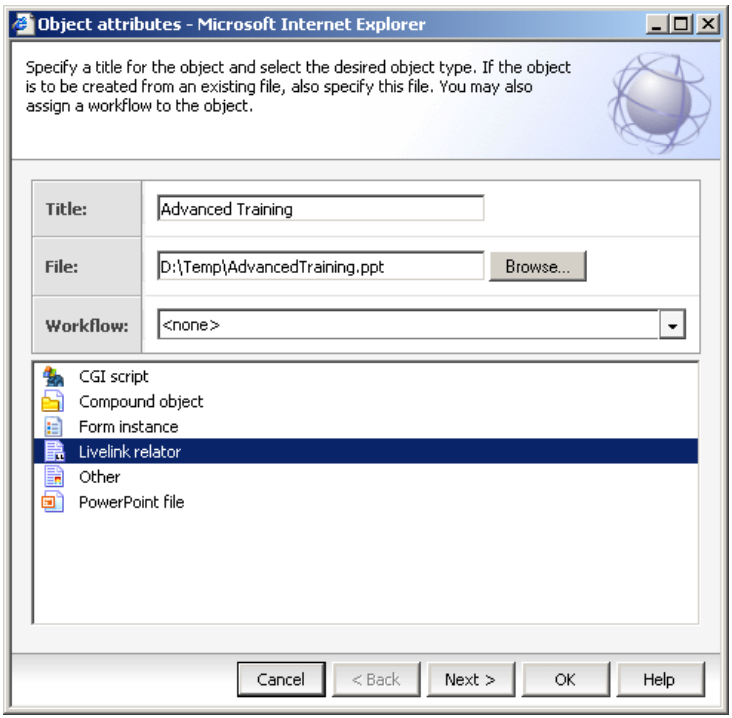


Fig. 107 – Specifying title, file, and object type

- *Title:* Enter the name for the document in Livelink. The Content client uses this name as the relator's title. The title is used to identify the relator and is displayed both in the header and in the navigation area.

  Entering a title is mandatory.

  > **Note:** Titles must not be more than 255 characters long. Also, you should adapt the length of the title in such a way that the resulting file name meets the requirements of your operating system. Under Windows, the absolute path to a topic and a generated page may not be longer than 240 and 260 characters respectively.

- *File:* Specify the document to be added to the Livelink system. Click the *Browse* button to select the file.

  Specifying a file is mandatory.

  > **Note:** Please choose only files that have a file extension.

- *Workflow:* If you want to assign a workflow to the new object, click the desired workflow in the drop-down list.
- Object type: Select the object type “Livelink relator”.

Confirm by clicking the *Next* button.

Specifying the Livelink system

There may be more than one Livelink system in your company. In the following dialog box, you select the Livelink system to which the document will be added.

Confirm by clicking the *Next* button.

Specifying location in Livelink

In the following dialog box, you select the Livelink folder for the new document.

Confirm by clicking the *Next* button.

Entering a remark

In the following dialog box, you may enter information for the relator's log.

Confirm your entries by clicking the *OK* button.

The WCM system creates the relator and shows it in the Navigation area. The relator's status is “changed”. Additionally, the document will be stored at the specified location in the Livelink system.

The relator is created with the settings *Update Edit version automatically* and *Delete Livelink object together with relator*. You can change these settings in the Livelink metadata dialog box (see section “Modifying Livelink Metadata” on page 355).

For the relator to become visible in the published website, you must submit it. After this, the relator must be released by Quality Assurance.

Editing Relators

Relators do not only enable you to publish Livelink objects on a website. They also make it possible to edit Livelink objects via the Content client.

In this section, the following procedures are introduced:

- editing the content of a relator, see the following section
- editing the Livelink metadata, see section “Modifying Livelink Metadata” on page 355
- integrating the Livelink metadata in a web page, see section “Integrating Livelink Metadata in a Page” on page 358
- manually synchronizing the relator when the Livelink object has been changed, see section “Updating a Relator Manually” on page 361

Modifying the Content of a Relator

Objective

Relators refer to objects in the Livelink system. You can use the Content client to modify the content of Livelink objects.

Prerequisites

- The relator has the status “changed”, “rejected”, “released”, or “delayed release”.
- The radio button *Update Edit version automatically* is selected for the relator.
- You have the access rights “Read” and “Change object” for the relator.
- You have the permission “Reserve” for the Livelink object.

Procedure

1. Change to the Edit view of the website.
2. In the Navigation area, mark the relator whose associated Livelink object you want to edit.
3. Choose *Edit* → *Check out* or click the appropriate icon.

   Icon for checking out an object

   The relator is barred for write access by other users; the Livelink object will be marked as reserved in the Livelink system.

   Depending on the configuration of the Content client, the content of the Livelink object opens in the integrated HTML editor or in the associated editor program.
4. Edit the Livelink object in the respective editor.
5. Save your changes and close the editor.

6. To return the modified relator to the WCM system and unreserve the Livelink object, choose *Edit* → *Check in* or click the appropriate icon.

   Icon for checking in an object

   The relator has the status “changed”. A new version is added to the Livelink object.

   For the changes to become visible in the published website, you must submit the changed relator which then must be released by Quality Assurance.

Modifying Livelink Metadata

Objective

Each Livelink object has a number of metadata. That means that relators have Livelink metadata in addition to the WCM metadata, such as Livelink system and Livelink ID. You can view and modify the Livelink metadata of relators in the Livelink metadata dialog box.

You want to edit the Livelink metadata of a relator.

Prerequisites

- The relator has the status “changed”, “rejected”, “released”, or “delayed release”.
- You have the access rights “Read” and “Change metadata” for the relator.
- You are assigned the functional area “Livelink”. To check your functional areas, choose *Tools* → *Login info*.
- You have the permission “Modify” for the Livelink object.

Procedure

1. Change to the Edit view of the website.
2. In the Navigation area, mark the relator whose Livelink metadata you want to edit.
3. Choose *View* → *Livelink metadata*.

Fig. 108 – The Livelink metadata dialog box

The dialog box provides several tabs for viewing and editing metadata.

- *General Livelink data* tab: Most entries on this tab are controlled by the Livelink system and can only be viewed here.

  In the fields *Description* and *Version description*, you may enter basic information on the relator and on this special version of the relator.

  In the *Version* area, you determine whether the relator is to be updated automatically when the associated Livelink object changes (*Update Edit version automatically* radio button) or is to reference a specific version of the Livelink object (*Fixed version* radio button).

  Select the *Delete Livelink object together with relator* check box to determine that the referenced Livelink object will be destroyed together with the relator.

- Tabs for object categories: For each object category assigned to the object in the Livelink system, a separate tab with the category's name is shown (in the figure *Training Materials*). You can modify the Livelink object's attributes on the tabs.

4. Make the desired changes on the tabs.
5. Click the *Save* button.

   The metadata are changed in the Livelink system. The relator now has the status “changed”.

   For the changes to become visible in the published website, you must submit the changed relator which then must be released by Quality Assurance.

Integrating Livelink Metadata in a Page

Just like WCM metadata, you can integrate Livelink metadata in a template or in the page generated for the object and thus use the metadata in a WCM-managed website. For integrating metadata, WCM tags are provided.

> **Note:** For detailed information on WCM tags, refer to chapter 14 “Mit WCM-Tags arbeiten”.

Livelink Standard Metadata

The following table provides an overview of the WCM tags for the most common Livelink standard metadata. The file **Livelink_WCM_tags.html** in the directory **{WCM installation directory}\examples\wcmtags** contains all Livelink-specific WCM tags.

Table 23 – WCM tags for the most common Livelink metadata

Metadata item	Internal name	WCM tag
Description	`comment`	`{VIPLL attribute="comment"}`
File name	`filename`	`{VIPLL attribute="filename"}`
Created by	`createdby`	`{VIPLL attribute="createdby"}`
Created	`createdate`	`{VIPLL attribute="createdate"}`
Modified	`modifydate`	`{VIPLL attribute="modifydate"}`
MIME type	`mimetype`	`{VIPLL attribute="mimetype"}`
Object size (in byte)	`filedatasize`	`{VIPLL attribute="filedatasize"}`

Metadata item	Internal name	WCM tag
Node ID	nodeid	{VIPLL attribute="nodeid"}
Version	versionnum	{VIPLL attribute="versionnum"}
Version description	vcomment	{VIPLL attribute="vcomment"}

User-Defined Metadata

The WCM tag mechanism is also available for user-defined Livelink metadata (e.g. the attributes of an object category). Use the following pattern to form WCM tags for user-defined attributes:

```
{VIPLL category="<category name>" set="<set name>" attribute="<attribute name>" set_no="<number>" item_no="<number>"}
```

- `category`: name of the category containing the attribute
- `set`: name of the set containing the attribute
- `attribute`: name of the attribute for which the WCM tag is created
- `set_no`: position of the set in the list of set values
- `item_no`: position of the attribute value to be shown in the list of attribute values

If the attribute to be shown is not of type “set”, the expression becomes simpler:

```
{VIPLL category="<category name>" attribute="<attribute name>" item_no="<number>"}
```

Example

For the training materials, you created a category named “Training Materials”. By means of the attributes of this category, you can manage information on the status and the contact person for the materials. The attribute set “Contact Information” of the category contains three attributes in the following order:

1. Contact Person
2. E-mail
3. Phone number

   The attribute “Phone number” may contain two values: in the first position the conventional phone number, in the second position the mobile phone number.

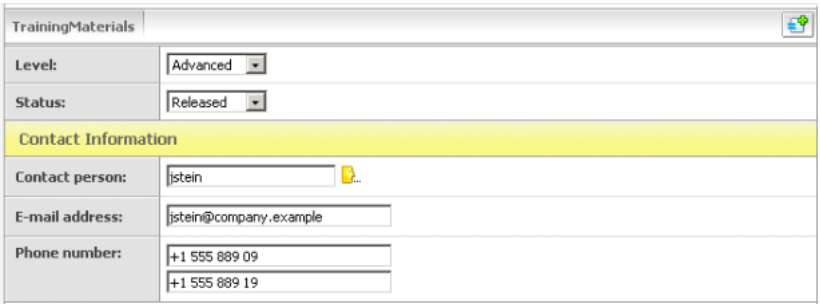


Fig. 109 – The category “training materials” in the Livelink system

Use the WCM tag

```
{VIPLL category="Training Materials" set="Contact Information"
attribute="Phone number" set_no="1" item_no="2"}
```

to show the mobile phone number of the contact person on the page. In case of the employee “Joshua Stein” who is entered as the contact person, this is the phone number “+1 898 999 333”.

Updating a Relator Manually

Objective

If you have selected the *Fixed version* radio button for the relator, the relator will **not** be updated automatically when the referenced Livelink object changes. You want to update the relator manually.

Prerequisites

- The relator has the status “changed”, “rejected”, “released”, or “delayed release”.
- The *Fixed version* radio button was selected for the relator.
- You have the access rights “Read” and “Change metadata” for the relator.
- You are assigned the functional area “Livelink”. To check your functional areas, choose *Tools* → *Login info*.

Procedure

1. Change to the Edit view of the website.
2. In the Navigation area, mark the relator that you want to update.
3. Choose *View* → *Livelink metadata* → *General Livelink data* tab.
4. In the *Version* drop-down list, click the version of the Livelink object that the relator is to reflect.
5. Click the *Save* button.

   For the changes to become visible in the published website, you must submit the changed relator which then must be released by Quality Assurance.

Deleting a Relator

Notes

- You can delete relators like all other “normal” WCM objects. For detailed information on deleting WCM objects, refer to section “Objekt löschen” starting on page 216.
- **Delete Livelink object together with the relator**: In order to destroy the Livelink object, you require the Livelink permission “Delete”. If you do not have this permission, the relator will be deleted in the WCM . However, the object in the Livelink system will not be removed.
- **Livelink relators** are always deleted when the referenced objects are deleted in Livelink. Relators that have the status “submitted” are automatically rejected and deleted afterwards. If a relator has already been released once, it is marked as deleted and must be destroyed manually.
- **Livelink folder relators** for which the *Update Edit version automatically* option is selected, are automatically deleted (including all subordinate objects), when the referenced Livelink folder is deleted.

  If you want to add further WCM objects below the Livelink folder relator, you should click the *Fixed version* radio button for the folder relator. Folders with this settings will not be removed when the referenced Livelink folder is deleted.

Selecting the *Delete Livelink object together with relator* option

In the Livelink metadata dialog box, you can determine that the referenced Livelink object will be destroyed together with the relator.

Proceed as follows:

1. Change to the Edit view of the website.
2. Open the Livelink metadata dialog box by choosing *View→ Livelink metadata*.
3. On the *General Livelink data* tab, select the check box *Delete Livelink object together with relator*.
4. Click the *Save* button.

   For the change to take effect, you must submit the relator which then must be released by Quality Assurance.

CHAPTER 12

Finding Objects and Editing Them Together

The Content client provides extensive filter functions in order to find objects quickly. You can search objects according to certain criteria and perform actions such as “Submit” or “Release” on the objects found. This way, you can edit many objects simultaneously. This considerably speeds up work procedures and makes them more efficient, as it is also possible to send compact e-mails to the responsible editors and QA staff via the filter navigation.

The following filter functions are available in the Content client:

- The **object filter** contains predefined and ready-to-use standard filters (see the following section).
- The **“My objects”** view contains predefined and ready-to-use standard filters that take account of your access rights. The hitlist is personalized, i.e. only those objects that you have edited or that you may edit are available to you (see the following section).
- The **“My work list”** view offers you filters for searching WCM objects that are in a certain workflow activity or to which a certain workflow is assigned. The results are personalized. Only those objects that you may edit are displayed. The available filters depend on the workflow definitions in your WCM system (see section “The “My worklist” view” on page 98).

- The **filter editor** enables you to create any number of filters according to custom criteria. This is especially useful in the case of recurring and complicated queries (see section “The Filter Editor” starting on page 376).

The Standard Filters

> **Note:** To be able to use the standard filters, you require the functional areas “Filter standard” and “View My objects”. To check your functional areas, choose *Tools* → *Login info*.

In this section, you get information on the following aspects of the standard filters:

- using the standard filters, see the following section
- description of the filter procedure on the basis of two application examples, see section“Examples of Using the Standard Filters” starting on page 373

Using the Standard Filters

The standard filters enable you to search for objects on the basis of their object status. Additionally, there are a number of other search options, such as references between objects.

Opening the Object Filter or “My objects” View

In the Navigation area, mark the topic in which you want to search.

Choose *View* → *Object filter* to open the object filter.

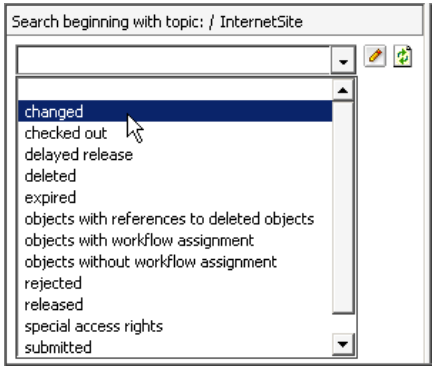


Fig. 110 – Standard filters of the object filter

Choose *View* → *My objects* to open the “My objects” view.

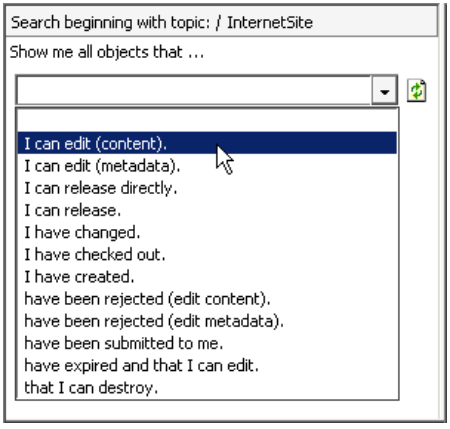

Search beginning with topic: / InternetSite
Show me all objects that ...
I can edit (content).
I can edit (metadata).
I can release directly.
I can release.
I have changed.
I have checked out.
I have created.
have been rejected (edit content).
have been rejected (edit metadata).
have been submitted to me.
have expired and that I can edit.
that I can destroy.

Fig. 111 – Standard filters of the “My objects” view

The drop-down list contains the predefined standard filters. The path to the topic that you marked in the navigation is displayed above the lists.

Setting the Filter Criterion

Click the desired filter criterion in the drop-down list of the object filter or the “My objects” view. In the configuration of the Content client, you can determine which filters are displayed in the drop-down lists.

The following tables explain the standard filters.

Table 24 – Standard filters in the object filter

Filter	Description
changed	Finds objects with the status “changed”
checked out	Finds objects with the status “checked out”
delayed release	Finds objects with the status “delayed release”
deleted	Finds objects with the status “deleted”
expired	Finds objects that have exceeded their expiration date
objects with references to deleted objects	Finds objects that have invalid references because they reference deleted objects
objects with workflow assignment	Finds objects to which a workflow is assigned
objects without workflow assignment	Finds objects to which no workflow is assigned
rejected	Finds objects that have been rejected by Quality Assurance and sent back to the responsible editor for correction
released	Finds objects with the status “released”
special access rights	Finds objects that do not inherit their access settings from the parent topic, i.e. objects that have an access control list of their own

Filter	Description
submitted	Finds objects that have been submitted to Quality Assurance
without filter	Searches all objects in the marked topic and below

Table 25 – Standard filters in the “My objects” view

Filter	Description
Show me all objects that ...	
I can destroy.	Finds objects with the status “deleted” for which you have the access right “Release”
I can edit (content).	Finds objects with the status “released”, “changed”, or “rejected” for which you have the access right “Change object”
I can edit (metadata).	Finds objects with the status “released”, “changed”, or “rejected” for which you have the access right “Change metadata”
I can release directly.	Finds objects with the status “changed” for which you have the right “Change object” and for which the “Direct release” check box has been selected in the Metadata dialog box. This setting must have been released by Quality Assurance.
I can release.	Finds objects with the status “submitted” for which you have the access right “Release”
I have changed.	Finds objects with the status “changed” that have been edited last by you (the value of the metadata item ”Modified by“ is your user ID) or that have recently been created by you

Filter	Description
I have checked out.	Finds objects that have been checked out by you
I have created.	Finds objects that have been added by you
have been rejected (edit content).	Finds objects with the status “rejected” for which you have the access right “Change object”
have been rejected (edit metadata).	Finds objects with the status “rejected” for which you have the access right “Change metadata”
have been submitted to me.	Finds objects with the status “submitted” or “deleted” for which you have the access right “Release”
have expired and that I can edit.	Finds objects that have exceeded their expiration date, that are in the status “released”, “changed”, or “rejected”, and for which you have the access rights “Change object” and “Change metadata”

Click the desired filter in the drop-down list.

The filter hitlist opens automatically.

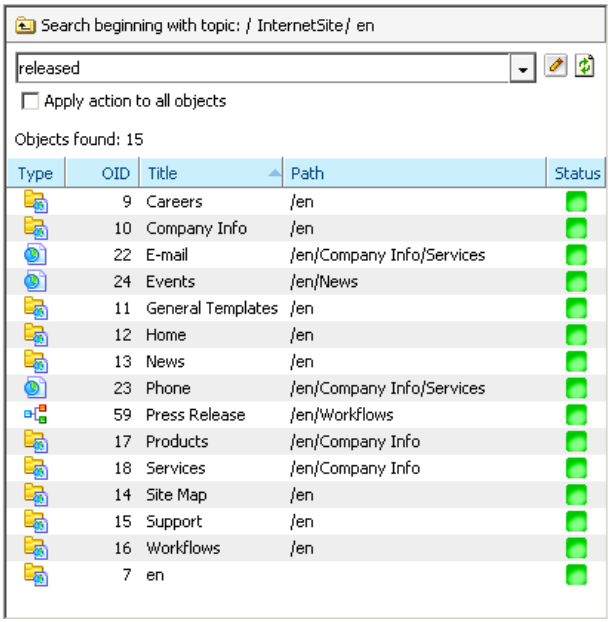


Fig. 112 – Filter hitlist

For each object, the object type, OID, title, path to the root object, and status are displayed. By clicking on the header row of a column, you can sort the hitlist by the criterion in this column. For example, clicking on the *Title* column sorts the list by title in ascending alphabetical order. To reverse the sort order, click the header row again. This way you can clearly arrange long lists.

You can determine the maximum number of objects to be displayed in the filter hitlist. If the number of objects found exceeds the value specified, the hitlist is displayed on several pages. By means of a number bar below the hitlist, you can browse through the individual pages.

To apply the filter again, click the *Apply filter again* icon in the navigation area.

Icon for applying a filter again

To repeat the search for a topic located higher in the object hierarchy, click the icon above the drop-down list or the respective entry in the path. By double-clicking a topic in the hitlist, you repeat the search in this topic.

Selecting Objects for Editing

You can mark one or more objects in the list in order to edit them.

If you want to edit all objects found in the object filter, select the check box *Apply action to all objects*.

Editing Selected Objects

To edit the marked objects, choose the respective command from the menu or click the appropriate icon in the toolbar. e.g. *Submit*.

For each object selected, Livelink WCM Server checks whether the action can be performed. This depends on the following factors:

- the object status of the individual object
- your access rights for the individual objects
- certain object properties, such as “Direct release”

Notes

- Objects on which the action cannot be performed are removed from the selection.
- If an error occurs while the action is being performed, the entire action is canceled. Changes that have already been made are undone.

- You can use the filter while an asynchronous action is in progress. If the filter hitlist contains objects that are also affected by the asynchronous action, the data of these objects (e.g. the status) might not be up-to-date. This might lead to errors if you perform an action on the filtered objects.
- A filter hitlist may contain a compound object with its dependent objects. You should not perform any actions on these objects as this may lead to errors in the Content client.

Examples of Using the Standard Filters

In the following, the use of the standard filters in the “My objects” view and in the object filter is described on the basis of a simple example.

Example: Submitting Changed Objects

Objective

You want to submit all objects that you have changed in the topic “Products” to Quality Assurance.

Prerequisites

- The objects have the status “changed”.
- You have the access rights “Read” and “Change object” for the objects.
- You are assigned the functional area “View My objects”. To check your functional areas, choose *Tools* → *Login info*.

Procedure

1. Change to the Edit view of the website.
2. In the navigation area, mark the topic “Products”.

3. Choose *View* → *My objects* to open the “My objects” view.
4. In the drop-down list, click the filter *I have changed*.

   The hitlist displays all objects that you have changed in the topic “Products”.
5. As you want to submit all objects that you have changed, mark all objects in the hitlist.
6. Choose *Object* → *Submit* or click the appropriate icon.

   Icon for submitting objects

   If the Content client is configured accordingly, an E-mail dialog box opens. In this dialog box, you can submit the objects and send a message to the e-mail address saved in the metadata item “E-mail QA”.
7. Click the *OK* button.

   The system checks whether all marked objects meet the requirements. Objects for which the action “Submit” cannot be performed are removed from the selection. The other objects acquire the status “submitted” and can be checked by Quality Assurance.

Example: Releasing Objects

Objective

You want to release all submitted objects.

Prerequisites

- The objects have the status “submitted”.
- You have the access rights “Read” and “Release” for the objects.
- You are assigned the functional area “Filter standard”. To check your functional areas, choose *Tools* → *Login info*.

Procedure

1. Choose *View* → *Object filter* to open the object filter.
2. In the drop-down list, click the filter *submitted*.

   The hitlist displays all objects that have been submitted.
3. In the hit list, select the objects individually and check their quality. To view the submitted objects in the right window pane, choose *View* → *Content (WYSIWYG)*.
4. In the hitlist, mark the objects that you want to release.
5. Choose *Object* → *Release* or click the appropriate icon.

   Icon for releasing objects

   If the Content client is configured accordingly, an E-mail dialog box opens. In this dialog box, you can release the objects and send a message to the e-mail address saved in the metadata item “E-mail Release”.
6. Click the *OK* button.

   The system checks whether all marked objects meet the requirements. Objects for which the action “Release” cannot be performed are removed from the selection without a message being displayed. The other objects acquire the status “released”. This makes the released version of the objects available in the Production view.

The Filter Editor

> **Note:** To work with the filter editor, you require the functional areas “Filter standard” and “Filter edit”. To check your functional areas, choose *Tools* → *Login info*.

The filter editor helps you create custom filters with freely definable criteria. Stored filters are added to the drop-down list of the object filter. In contrast to the standard filters, you can edit and delete user-defined filters.

To open the filter editor:

1. Choose *View* → *Object filter* to open the object filter.
2. Click the *Filter editor* icon in the navigation area.

   Icon for opening the filter editor

   The filter editor opens in a separate window.

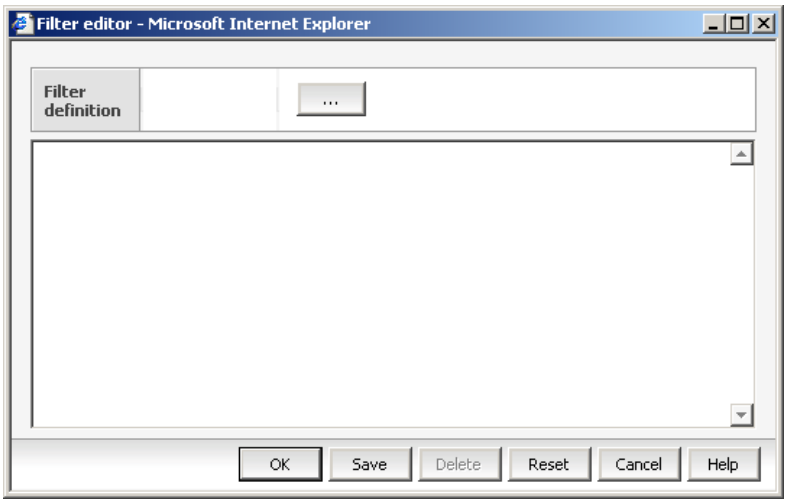


Fig. 113 – The filter editor

The following functions are available to you:

- creating user-defined filters by means of a wizard, see the following section
- changing user-defined filters, see section “Editing Custom Filters” starting on page 383
- deleting user-defined filters, see section “Deleting a Custom Filter” starting on page 385
- using the filter expression language for creating user-defined filters, see section “The Filter Expression Language” starting on page 385

Creating a Custom Filter

A wizard assists you in defining custom filters. Thus, it is not necessary to know the filter expression language in order to define filters.

Starting the Wizard

1. Choose *View* → *Object filter* to open the object filter.
2. Click the *Filter editor* icon in the navigation area.

   Icon for opening the filter editor

3. Click the button .

Specifying the Filter Criterion

Here you specify the metadata item that you want to use as the filter criterion.

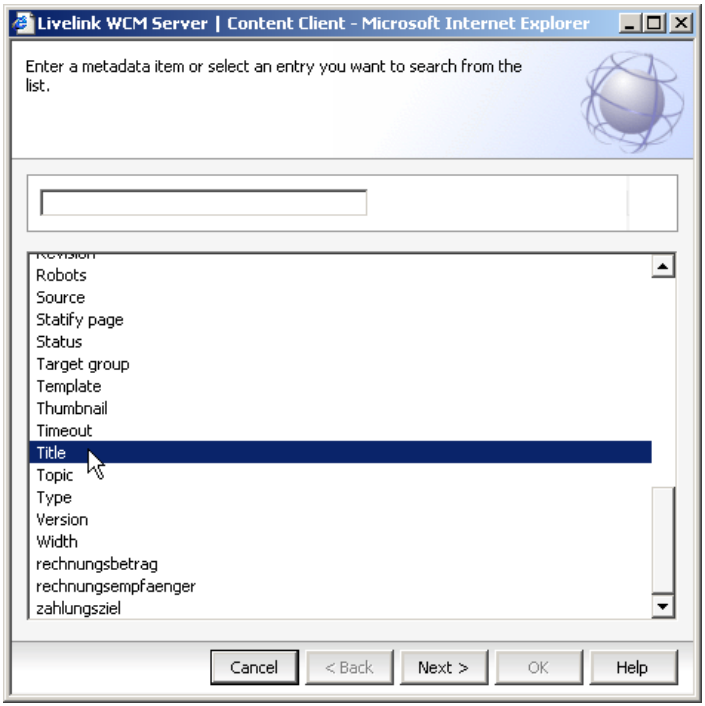


Fig. 114 – Specifying the Filter Criterion

The drop-down list in this dialog box contains the standard metadata. Select a metadata item from the list. To filter for special attributes, enter the name of the special attribute in the text field above the list.

Click the *Next* button.

Determining a Comparator

Here you determine a comparator. The comparator combines the metadata item defined in the second step with the value specified in the fourth step.

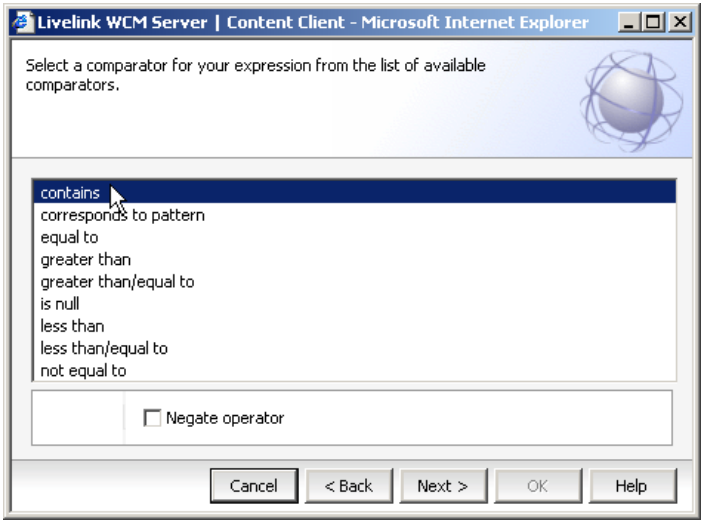


Fig. 115 – Determining a comparator

The selection list contains only those comparators that are applicable for the metadata item specified in the second step.

By means of the *Negate operator* check box, you can reverse the meaning of the operator. (Example: if you select the *Negate operator* check box for “contains”, this corresponds to “does not contain”.)

The individual operators are described in section “Comparators” on page 385.

Select a comparator from the list and click the *Next* button.

Specifying a Comparison Value

If (in the third step) you have selected a comparator (e.g. “not equal to”) that requires a comparison value, specify the value here.

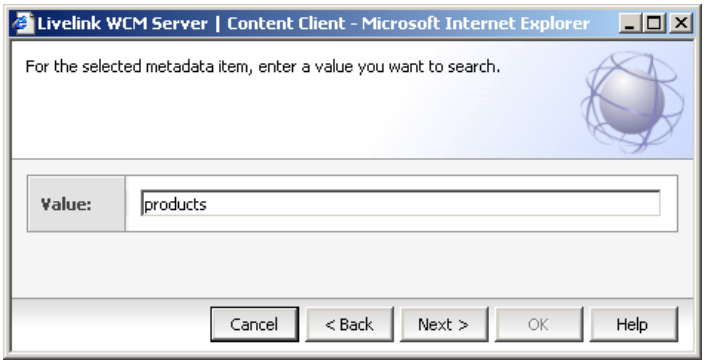


Fig. 116 – Specifying a comparison value

For more information on comparison values, see section “Comparison Values” on page 388.

Click the *Next* button.

Confirming the Filter Expression

The filter expression is constructed from your entries.

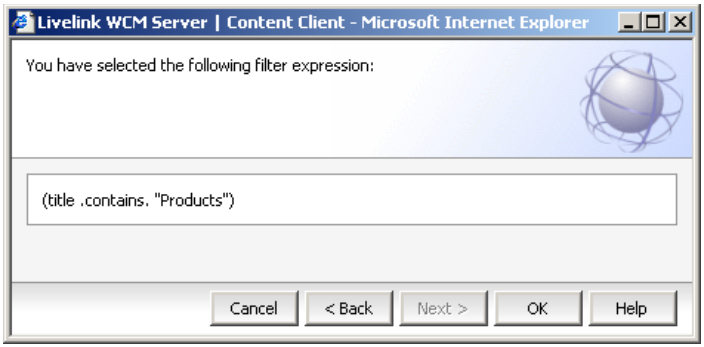


Fig. 117 – Generated filter expression

The filter expression displayed finds objects whose title contains the expression “Products”.

Click the *OK* button.

The filter expression is written to the filter definition field.

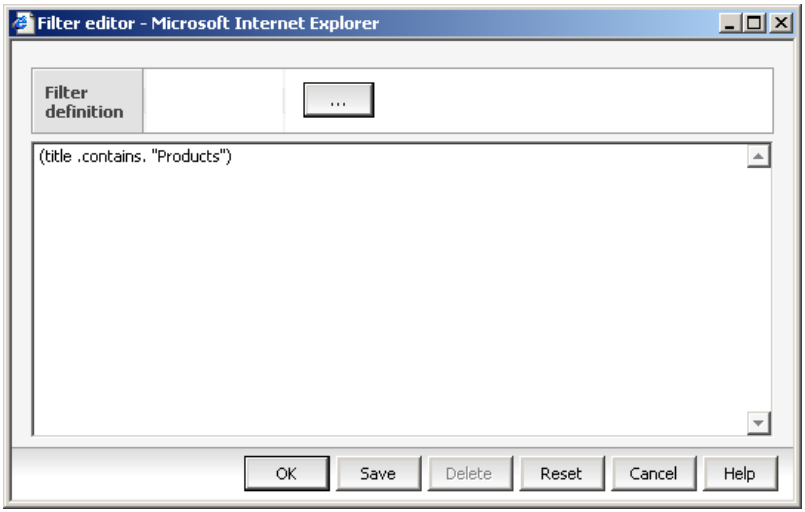


Fig. 118 – Adopting the filter expression

Saving and Applying the Filter

To apply the new filter, click the *OK* button. If you want to save the filter for future filter actions, click the *Save* button.

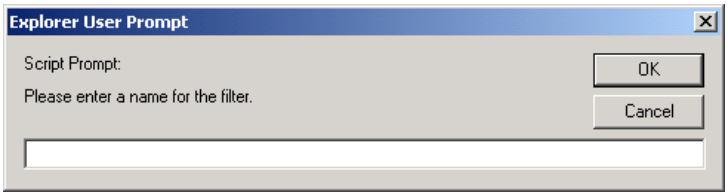


Fig. 119 – Saving a user-defined filter

Enter a name for the filter and click the *OK* button.

The filter is integrated in the drop-down list of the object filter and is applied automatically.

Filter Variables

Instead of specifying a concrete comparison value for a filter, you can use a variable for the comparison value. In this way, you can use the same filter for several queries. A filter variable has the format “${variable name}”, e.g. “$Search Term”.

Example

In section“Creating a Custom Filter” on page 377, a filter was defined which finds objects that contain the expression “Products” in their title. You can generalize this filter in such a way that it finds objects whose title contains a freely definable expression. To do so, replace the comparison value “Products” with the filter variable “$Search Term”.

When using this filter, you will see an input field for the filter variable in the navigation area. Enter the desired search term in this field.

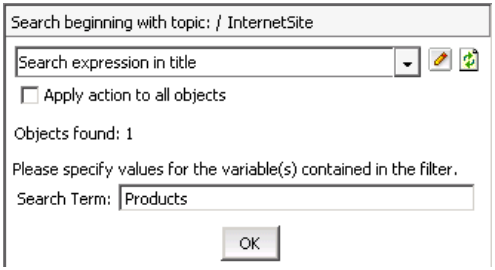

Search beginning with topic: / InternetSite
Search expression in title
Apply action to all objects
Objects found: 1
Please specify values for the variable(s) contained in the filter.
Search Term: Products
OK

Fig. 120 – Entering the value for the filter variable

Start searching by clicking the *OK* button. The search results are displayed on the *Filter result* tab. On the *Filter variables* tab, you can start a new search.

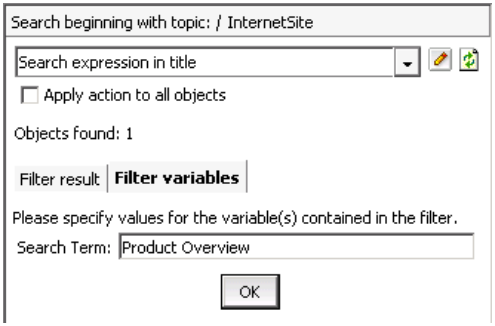

Search beginning with topic: / InternetSite
Search expression in title
Apply action to all objects
Objects found: 1
Filter result
Filter variables
Please specify values for the variable(s) contained in the filter.
Search Term: Product Overview
OK

Fig. 121 – Searching with a new search expression

Editing Custom Filters

In contrast to the standard filters, which cannot be changed, it is possible to edit user-defined filters.

To edit a user-defined filter:

1. Choose *View* → *Object filter* to open the object filter.
2. In the drop-down list, click the user-defined filter that you want to edit.

3. Click the *Filter editor* icon in the navigation area.

   Icon for opening the filter editor

4. Make the desired changes.

   To change the filter expression, make your changes directly in the filter definition. Example: to change the comparison value “Products” to “Support”, click the filter definition and replace “Products” with “Support”.

   To add expressions to the filter, click the button ... .

   The wizard for creating a user-defined filter starts with the following dialog box.

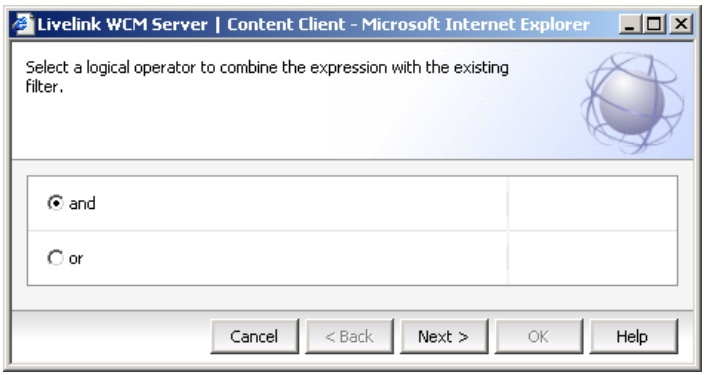


Fig. 122 – Defining a logical operator

   Mark the desired combination operator.

   The individual operators are described in section “Combination Operators” on page 390.

5. The following procedure corresponds to defining a new filter.

Deleting a Custom Filter

In contrast to the standard filters, which can only be hidden from the drop-down list via the Configuration dialog box, it is possible to delete user-defined filters.

> **Note:** Before deleting a filter, make sure that you do not use the filter within another filter (see section “Nested Filters” on page 392).

To delete a user-defined filter:

1. Choose *View* → *Object filter* to open the object filter.
2. In the drop-down list, click the user-defined filter that you want to delete.
3. Click the *Filter editor* icon in the navigation area.

   Icon for opening the filter editor

4. Click the *Delete* button.

   The filter is removed from the drop-down list in the object filter.

The Filter Expression Language

This section describes the individual elements of the filter expression language. The filter expression parser processes the filter expression language to construct the filters you have defined.

Comparators

The following table contains the operators that you can use to compare two values.

Table 26 – Comparators in the filter editor

Operator	Expression	Description	Can be applied to
equal to	=	As commonly used	All metadata
not equal to	!=		All metadata
less than	<		Character strings, numbers
less than/ equal to	<=	As commonly used	Character strings, numbers
greater than	>		Character strings, numbers
greater than/ equal to	>=		Character strings, numbers
corresponds to pattern	.like.	Compares patterns The comparison value can contain the percent character (%) as a placeholder. The percent character replaces any number of characters at the beginning, at the end, or in the middle of the comparison value.	Character strings
contains	.contains.	Finds character sub-strings	Character strings
is subordinate object of	.childof.	Finds subordinate objects This operator does not require a comparison value.	Metadata item “OID”

Operator	Expression	Description	Can be applied to
is null	`.isnull.`	Finds objects for which this metadata item is not set This operator does not require a comparison value.	All metadata

Notes:

For metadata with the data type “set” or “list”, you can only use the comparators “corresponds to pattern”, “contains”, and “is null”.

For finding a version, the operator `.version.` is available. This operator is combined with a comparator. This results in the following filter expression: `{version number} .version. {comparator}`. For example, to find objects with a version number less than 1.0.2, use the expression `1.0.2 .version. <`.

The resulting filter expressions must correspond to the following pattern:

- for operators requiring a comparison value:

```
{metadata item} {comparator} {value}
```

- for the operator “is subordinate object of”:

```
.childof. {OID}
```

- for the operator “is null”:

```
{metadata item} .isnull.
```

- for searching for a version:

```
{version number} .version. {comparator}
```

Comparison Values

Each WCM metadata item is assigned a certain data type that has a special format. When you specify a comparison value, you must consider the data type of the metadata item. The following table offers an overview of the data types and the associated formats.

Table 27 – Comparison values in the filter editor

Metadata item	Data type	Format
Modification date	Time stamp	YYYY-MM-DD HH:MM:SS or character string “now” **Example:** The date “September 30, 2003, 9:30” is entered as “2003-09-30 09:30:00”.
Expiration date		
Created on		
Release date		
Delayed release		
Author	User ID	Character string (e.g. “jstein”)
Released by		
Modified by		
Description	String	String
Direct release	Boolean value	Character string “true” or “false”

Metadata item	Data type	Format
Object category	String	String
OID	OID	Integer
Keywords	List	String
Language	Language	Locale (e.g. en_US for English)
Status	Status	String
Title	String	String
Type	Object type	String
Heading	String	String
Version	Version	Major.Minor.Micro (e.g. 5.0.3)
Template	OID	Integer
Target group	String	String

Character strings can be enclosed in double or single quotes. Quotes and backslashes (“\”) used within a character string must be preceded by a backslash. For clarification, the following table contains two examples.

Table 28 – Examples of the spelling of comparators

Comparison value	Spelling
The document “Products”	“The document \”Products\””
The directory \CompanyInfo\Products\	“The directory \\CompanyInfo\\Products\\”

Note: If you precede an metadata item of the data type “string”, “set”, or “list” with the operator `.ignorecase.`, then the comparison will not be case sensitive.

Combination Operators

The following table shows the operators that can be used to form a complex filter expression.

Table 29 – Combination operators in the filter editor

Operator	Expression	Description
and	`.and.`	As commonly used A combination with “and” takes priority over a combination with “or”. This can be changed by enclosing the expressions in brackets.
or	`.or.`	
not	`.not.`	Negates the filter expression

The filter expressions must correspond to the following pattern:

- for “and” and “or”:

```
{expression 1} {combination operator} {expression 2}
```

- for “not”:

```
.not. {expression}
```

Searching for Principals in Access Control Lists

The following table contains operators you can use to find objects whose access control list contains a certain principal.

For an object to be found, the following requirements must be met:

- The principal is explicitly assigned to the object, i.e. not only via the principal's group or role membership.
- The object has its own access control list; it does not inherit the access control list from the parent topic.

Table 30 – Principal operators in the filter editor

Operator	Expression
User in access control list	`.user-inacl.`
Group in access control list	`.group-inacl.`
Role in access control list	`.role-inacl.`

The filter expression must correspond to the following pattern:

```
{principal ID} {operator}
```

Example

Use the filter expression

```
jstein .user-inacl.
```

to search for objects whose access control list contain the user “jstein”.

Searching for Objects with Workflow Assignment

By means of the operator `.hasworkflow.`, you can filter for objects to which a workflow is assigned.

Nested Filters

The operator `.filter.` makes it possible to use a filter from the drop-down list in another filter. The filter expression must correspond to the following pattern:

```
.filter. "{name of the filter}"
```

If you want to use one of the standard filters within another filter, specify the internal name of the standard filter. The **standardfilter.xml** file in the directory **{WCM installation directory}\website\cmsclient\resource\config** contains the internal names.

Example

Use the filter expression

```
.filter. "filter-state-released" .and. title .contains. "products"
```

to search for objects that are in the status “released” and whose title contains the expression “products”. Within this filter expression, the standard filter “released” is used.

Examples for Filter Expressions

Table 31 – Examples for Filter Expressions

Filter expression	Description
`(oty_name = HTML)`	Finds objects of type “HTML page”
`(sta_name = checked_out)`	Finds objects with the status “checked out”
`(title .like. "WCM%")`	Finds objects whose title begins with “WCM”
`(title .contains. .ignorecase. "auto")`	Finds objects whose title contains the character string “auto”. Uppercase and lowercase letters are ignored.
`(price < 1000)`	Finds objects with the special attribute “price” of type “Integer” with a value less than 1000
`(date_expire < now)`	Finds objects that have exceeded their expiration date
`(date_expire > "2005-05-01")`	Finds objects expiring after May 1, 2005
`(.childof. 500)`	Finds objects subordinate to the topic with the OID 500
`(oty_name = HTML) .or. (oty_name = TOPIC)`	Finds objects of type “HTML” page or “Topic”
`(created_by = jstein) .and. (sta_name = edited)`	Finds objects created by “jstein” with the status “changed”

Notes on Filtering

- You can directly release an object if the “Direct release” check box is selected for the object and this setting has been released by Quality Assurance. Thus, a filter searching for objects that can be released directly (filter expression: `direct_release = "true"`) only finds objects for which this setting has been released by Quality Assurance.
- By marking an object in the navigation area, you can determine the start object for filtering. However, this selection will **not** be considered if the filter contains one of the following expressions:
  - `.childof.` (is subordinate object of)
  - `topic = "..."` (topic equal to)
  - `oid = "..."` (OID equal to)

CHAPTER 13
Working with Templates

The basic concept behind the structure of WCM objects is the separation of layout and content. Each page in a website can be assigned a defined template, which can, however, be changed at any time. The template contains the layout information for the page. Changes to a template result in fully automatic changes to the layout of all pages based on that template. Thus, you only need to change the layout in the template to update the relevant pages in the entire website. This makes it easier to implement your style guide and ensure a consistent appearance for the pages in your website.

A template is a freely designable page that contains design elements (e.g. graphics, background image, background color, text and link colors) and functional elements (e.g. JavaScript). The template also provides placeholders – so-called WCM tags – for the actual content of the objects based on the template and for information from the object metadata. For detailed information on using WCM tags, refer to chapter 15 “Working with WCM Tags”.

Thus, creating a template means constructing a layout that contains not only defined design elements, but also content and metadata positioned by means of placeholders. Furthermore, navigation can also be implemented in the template, which means that editors are relieved of this task and are free to concentrate on the content of the objects.

Notes:

You cannot use framesets in a template.

An object using the Unicode character set can only be assigned templates that also use Unicode.

For objects to which a template is assigned, the head element is deleted and replaced with the head element of the template each time the object is checked in.

It is possible that both the body element of the template and the body element of the object contain attributes. When the page is generated – i.e. when template and object are merged – only the attributes in the body element of the template are considered. The attributes in the body element of the object are only considered if the body element of the template does not contain any attributes. The following table illustrates this.

Table 32 – Body tag when combining object and template

Object (body element)	Template (body element)
Page (body element)	
`<body bgcolor="#FF0000">`	`<body vlink="#FF0000">`
`<body vlink="#FF0000">`	
`<body bgcolor="#FF0000">`	`<body>`
`<body bgcolor="#FF0000">`	

This chapter explains it is explained how to use templates in Livelink WCM Server. It is described how to

- add templates in the Content client, see the following section
- edit and delete templates in the Content client, see section “Editing and Deleting Templates” starting on page 400
- work with cascading templates, see section “Cascading Templates” starting on page 401

Adding Templates

Objective

You want to add a template for WCM objects.

Prerequisites

- You have the access rights “Read” and “Create” for the topic to which you want to add the template.
- You are assigned the functional area that is linked with the object type of the new template. To check your functional areas, choose *Tools* → *Login info*.

Procedure

1. Change to the Edit view of the website.
2. In the navigation area, mark the topic to which you want to add the template.

> **Note:** You can add templates anywhere you like in the navigation tree. The clearest approach is to add the templates to the topic that contains the objects that are to be based on the template.

3. Choose *Object* → *New* or click the appropriate icon.

   Icon for adding an object

   The wizard for adding objects opens.

4. Make the necessary entries in the fields.

   To create a template, select a template object type such as “HTML template”.

   The remaining fields and the use of the wizard are described in section “Adding Objects on the Basis of an Object Type” starting on page 134.

5. Click the *OK* button.

   The new object is created by the system and displayed in the status bar with the status “changed”. It can be selected in the wizard for adding objects in the *Template* field.

   Note: The new template should be submitted to Quality Assurance and released before it is assigned to an object.

Editing and Deleting Templates

Templates are also subject to the predefined staging steps. A template – like any other object – is edited in the Edit view. The changed template must be submitted to Quality Assurance, where it is released or rejected and sent to the person responsible for correction.

Note: Changing a template affects all objects using this template. When a change is released by Quality Assurance, all objects based on that template are changed automatically. This process may be very time-consuming.

You cannot delete a template that is used by other objects. If you want to delete a template, you must make sure that this template is not assigned to an object. Use the “Template structure” view to check this.

Cascading Templates

Cascading templates make it possible to combine various templates within a website to form nested structures. This enables you to build up a hierarchical template structure in which the individual templates are the dependent objects of a higher-level template.

The advantage of cascading templates is that the basic features of all pages in a website (such as company logo, text color, background color, layout rules) only have to be defined once in the master template. You can then derive the templates of the individual topics or object types from the master template. If you change the master template, this change automatically takes effect in the dependent templates and hence in the objects that use the dependent templates.

By using the Metadata dialog box, it is possible to assign objects to a (different) template. You can do this with cascading templates as well. It is also possible to add templates that are not cascading, i.e. that are not themselves based on another template.

Structure of a Cascading Template

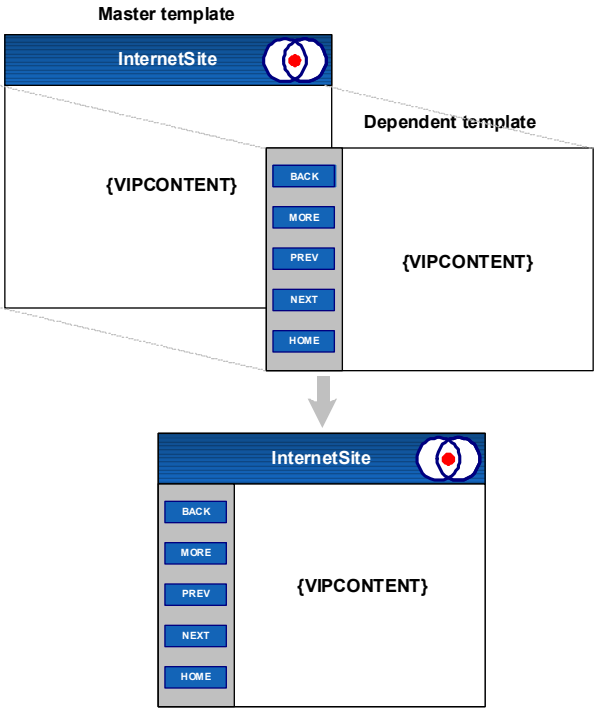


Fig. 123 – Example of a cascading template

Head element

In the head element of the cascading template – or to be more precise, in the combined (generated) template file – the settings from the head element of the master template and the dependent template are combined. First, all entries from the master template are adopted. Then the entries of the dependent template are inserted. In this way, different layout settings can be realized in the dependent template as the respective entries of the master template are overwritten.

```
<head>
  <!--  Settings from the master template -->
    <title>{VIPTITLE}</title>
  <!--  Settings from the dependent template -->
...
</head>
```

Note: When designing a template cascade, make sure the same attribute (e.g. `<title> ... </title>`) is not defined in the head element of several templates. This might cause repeated entries in the head element of the generated template when the head elements from the templates are combined. In this case, you cannot control which attribute content is displayed in the browser.

Body element

The body element of the cascading template is – as for other objects – formed by inserting the complete body element of the dependent template in the body element of the master template instead of the WCM tag `{VIPCONTENT}`.

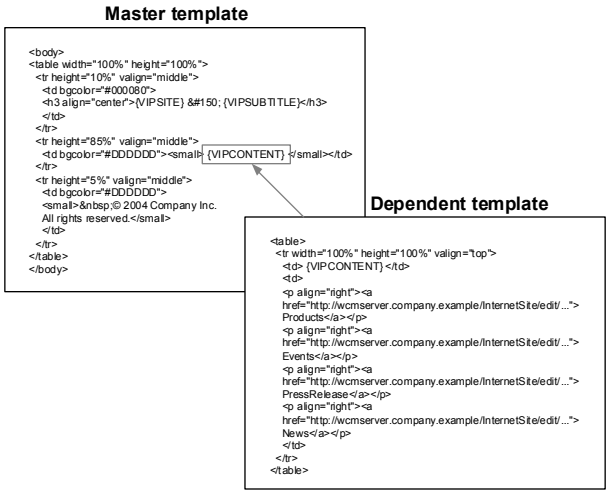


Fig. 124 – Merging the body element in a cascading template

When assembling the cascading template – unlike other object types – settings in the body element of the dependent template are also taken into account. For example, if the body element of the dependent template contains attributes (such as the attribute `bgcolor` to define the background color), these are copied into the generated template.

As a result of combining the two templates, it may happen that an attribute in the body element is present more than once, for example if a background color was defined in both templates. In that case, the setting in the dependent template takes priority and is copied into the generated template.

Adding Cascading Templates

To create a cascading template, first add an object in any topic you like. For cascading templates, make the following settings in the wizard for adding objects:

- Select a template object type, e.g. “HTML template”.
- In the window with the template structure, select the desired master template or enter the respective OID in the *Template* field.

Editing and Releasing Cascading Templates

When working with cascading templates, it is important to note the following points: changes to a master template do not take effect immediately in the dependent templates (and the objects that use the dependent templates). The dependent templates are not regenerated automatically. They are only generated when the master template is released.

> **Note:** If the master template has not been released yet – i.e. has the major version number 0 – the changes immediately take effect in the directly dependent templates. The other templates in the cascade will only be modified after the changed template has been released.

To check the effects of a master template that has been changed, but not yet released:

1. Change to the Edit view of the website.
2. Choose *View* → *Template structure*.

   In the template structure, you can see which templates and objects are dependent on the modified master template and are therefore affected by the change.
3. In the navigation area, mark a dependent template whose layout you want to change.

4. Choose *Object* → *Regenerate page*.

   If the Content client is configured accordingly, the dependent template is displayed on the basis of the changed master template.

   Note: If you use a dynamic deployment system, open the Configuration dialog box, choose *Dynamic deployment*, and click *Current template* in the *Template view* drop-down list.

   You can repeat the process for the other dependent templates and objects.

Deleting Cascading Templates

You can only delete a template from a cascading template structure if there are no dependent templates based on the template that you want to delete. There are two ways of ensuring this.

Deleting all dependent templates

First delete all templates that are dependent on the template that you want to delete. This deletion must be confirmed by Quality Assurance to ensure that the dependent templates are also deleted from the Production view. Only then is it possible to delete the parent template.

Isolating the template to be deleted

You can use the Metadata dialog box to change the assignment of a template to another template. In order to delete a template from the template cascade, you must isolate it from the cascade.

Example: The template “Department” is based on the template “DepartmentStore”, which is based on the template “Company”. In order to isolate the template “DepartmentStore”, assign the template “Company” to the template “Department” in the Metadata dialog box. Once the change to the metadata of the template “Department” has been released by Quality

Assurance, the template “DepartmentStore” is isolated from the template cascade. There is no template that is dependent on the template “DepartmentStore”. You can now delete this template (see the following figure).

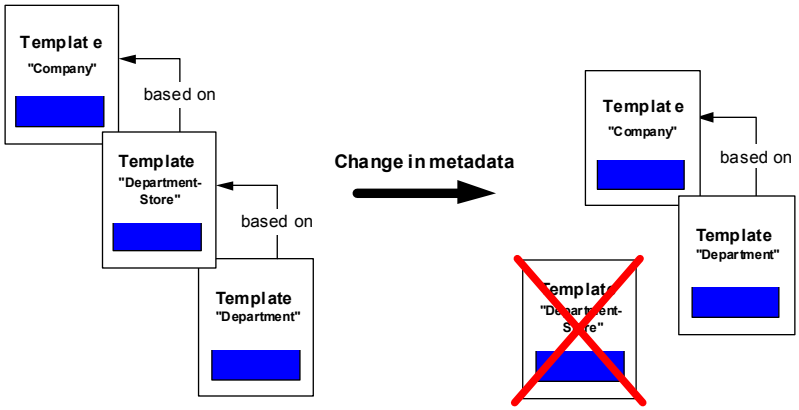


Fig. 125 – Deleting a template from a cascade

CHAPTER 14
Working with Special Attributes

In addition to the standard metadata, further information can be managed for each object. This object information is saved in the so-called special attributes. Special attributes supplement the metadata of an object. You can edit special attributes just like standard metadata.

Livelink WCM Server distinguishes the following two types of special attributes.

Attributes from the attribute set

Attribute sets contain attributes that describe special features of object types. Attributes may, for example, be the resolution of graphics, the associated editor, or a copyright notice. When adding a WCM object, you associate the object with an object type. As a result, the new object automatically acquires the attributes that this object type possesses. Like all metadata, attributes can be integrated in a website by means of WCM tags.

The object types and attribute sets themselves are managed via *Tools* → *Website administration* or in the Admin client; object types and attribute sets may be added or deleted there. Thus, attribute sets can be modified and are not rigidly defined by the system.

For further information on using attribute sets, refer to the Livelink WCM Server Administrator Manual. For an overview of the supplied attributes, refer to table “WCM tags for the supplied special attributes” on page 433.

Properties from the object category

By means of object categories, you can categorize WCM objects according to their content. Examples can be object categories named “Invoice” or “Article_description”. Object categories are defined by a number of properties, which can be freely specified via *Tools* → *Website administration* → *Object categories* or in the Admin client. An invoice contains, for example, the information on who will receive the invoice and which amount is to be paid when. Accordingly, the object category “Invoice” is characterized by the properties “invoice_recipient”, “invoice_amount”, and “payment_deadline”.

When adding a WCM object, you can optionally associate the object with an object category. It then possesses the properties that have been defined for this category via *Tools* → *Website administration* → *Object categories* or in the Admin client. By means of WCM tags, you can embed these properties in websites and use them when searching a website. For example, this makes it possible to show a customer all outstanding invoices.

In this chapter, you get information on the following aspects of special attributes:

- representation of special attributes in the Metadata dialog box, see the following section
- filtering for special attributes, see section “Special Attributes in the Object Filter” starting on page 412
- statification of objects, see section “Statification” starting on page 413
- activating thumbnails for WCM objects, see section “Activating Thumbnails for Objects” starting on page 422

Note: For detailed information on integrating special attributes by means of WCM tags, refer to chapter 15 “Working with WCM Tags”.

Special Attributes in the Metadata Dialog Box

The special attributes of an object are managed on the *Properties* and *Attributes* tabs in the Metadata dialog box. Each attribute or property corresponds to a field in which you can store special – object-type specific or object-category specific – information about the object.

For information on how to use the Metadata dialog box, refer to section “Editing Metadata” starting on page 193.

Example

Via *Tools* → *Website administration* → *Object categories* or in the Admin client, the object category “Invoice” has been added with the properties “invoice_recipient”, “invoice_amount”, and “payment_deadline”. If you assign this object category to your object, the *Properties* tab of the Metadata dialog box will look as follows:

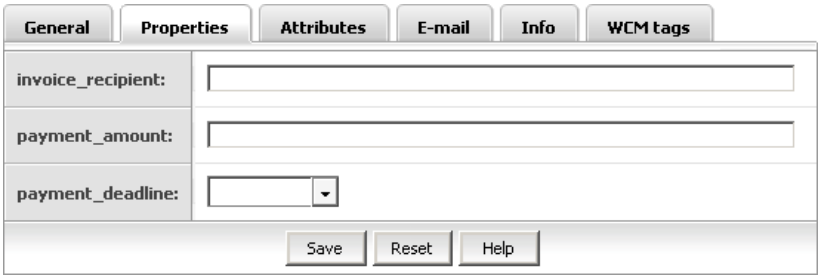


Fig. 126 – The object category “Invoice” in the Metadata dialog box

Here you can, for example, save the information on the deadline for paying the invoice. In the field of the property “payment_deadline”, enter the respective date.

Special Attributes in the Object Filter

By means of the filter editor, you can search for special attributes. These can be properties from the assigned object category or attributes from the attribute set assigned to the object type.

The following example explains how to filter for objects with the payment deadline Nov 30, 2003. The payment deadline is saved in the property “payment_deadline” from the object category “Invoice”.

To filter for the payment deadline Nov 30, 2003:

1. Choose *View* → *Object filter* to open the object filter.
2. Click the *Filter editor* icon in the navigation area.

   Icon for opening the filter editor

3. Click the button .
4. In the text field of the following dialog box, enter the special attribute “payment_deadline”.
5. In the following two dialogs boxes, enter the operator (“equal to”) and the date you want to search for.

   This results in the following filter expression:

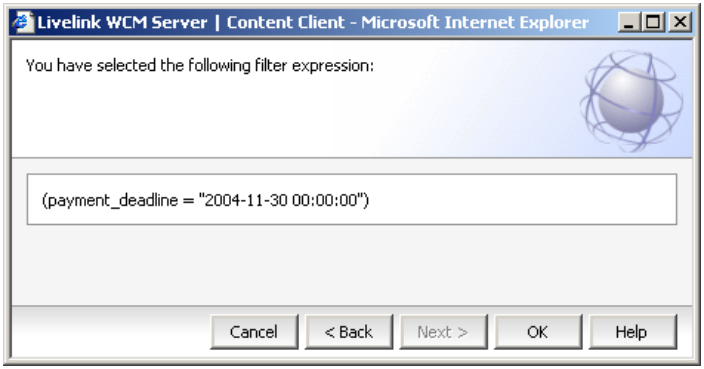

Livelink WCM Server | Content Client - Microsoft Internet Explorer
You have selected the following filter expression:
(payment_deadline = "2004-11-30 00:00:00")
Cancel
< Back
Next >
OK
Help

Fig. 127 – Filtering for the payment deadline Nov 30, 2004

6. Click the *OK* button.

Statification

In order to speed up access to dynamic content, it might be useful to convert the code of the dynamic page into static code. The term *statification* refers to the representation of dynamic content on static pages. When statification is performed, the code of the dynamic page is already executed during the (asynchronous) generation of the page, and the result – usually HTML code – is saved.

Statifying pages means freezing the object state and publishing it in the Production view of a website. For example, in the case of a page containing navigation elements, the current navigation status at the time of publication is recorded in the form of static elements on the page. Changes in the topic structure released subsequently are not reflected in the statified navigation. The dynamic page elements are updated only when the page is regenerated or republished. Statification is generally unsuitable for personalized pages, as these pages have to react dynamically, i.e. at runtime, to the user data entered.

Statification of pages depends on various conditions.

Configuration of the deployment system

- In the Admin client, you determine whether statification is to be performed for all objects of the website, for no objects, or depending on the object.
- In the Admin client, the check box *Analyze statified pages* can be selected. During this analysis, the OIDs for the WCM URLs used on the pages are saved in the database. If the URL of a referenced object changes, the respective pages are automatically regenerated and statified.

For detailed information on the configuration of deployment systems, refer to the online help of the Admin client.

Object type

Only objects whose object type is assigned to the attribute set “dynamic” can be statified. These are the following object types:

- “JSP page”, “JSP topic”, and “JSP template”
- “Form instance” and “Form template”
- “XML document”, “XML template”, “XSLT document”, and “XSLT template”

Note: Via *Tools* → *Website administration* → *Object types* or in the Admin client, it is possible to assign an object type (e.g. “Frame”), the attribute set “dynamic” and the generated file extension **jsp**. Such a “JSP frame object” can also be statified.

Special Attributes

The following attributes are important for statification of objects:

- **“Statify page”**
  The special attribute “Statify page” is used to control the object-dependent statification (see section “Object-Dependent Statification” on page 416).
- **“Timeout”**
  If the code of a dynamic page is to be converted into static code, the respective page is first generated as a temporary page. This temporary page is executed via a URL connection. The result of this execution is stored as a final static page.

  When configuring the deployment system in the Admin client, you set the waiting time (in milliseconds) for establishing the URL connection to the temporarily generated dynamic page. This value can be overwritten by means of the special attribute “Timeout” in the Metadata dialog box.

  The timeout value that applies to a page is determined as follows:

  1. The value of the special attribute “Timeout” is determined for the object to be statified.
  2. The value of the special attribute “Timeout” is determined for the template assigned to the object to be statified (if necessary, the template cascade is searched).
  3. The timeout value is read from the configuration of the deployment system.

  Thus, the timeout value specified in the Admin client is only used if neither the object to be statified nor one of its templates has a valid value for the special attribute “Timeout”.

> **Note:** Statification is performed by the Content server running in the context of a JSP engine or as web application in an application server. If this Content server is overloaded, the valid timeout value may be exceeded. This results in the abortion of the statification attempt and in error messages. Therefore, select generous timeout values.

- **“Absolute URL”**
  The special attribute “Absolute URL” plays, among other things, a special role when statifying form instances (see section “Statification of Form Instances” on page 418).

Object-Dependent Statification

An object whose type is associated with the attribute set “dynamic” possesses the special attribute “Statify page”. In the Metadata dialog box, you can use this attribute to determine how the code of the object and – if existing – of its template(s) is to be statified. This requires that object-dependent statification is set for the deployment system in the Admin client.

Statification of JSP Objects

Statification of JSP objects is object-dependent if the special attribute “Statify page” has the value `on` or `local`:

- `on`: This value causes the content of the object to be converted into static code. The same applies to all templates assigned to the object, provided their content is inserted into the page that the deployment system generates for the object.

- `local`: Regarding the body element, this value causes only the content of the object to be converted into static code. Additionally, the content of the entire head element – which may consist of the merged content of the head elements of the template cascade – is converted into static code.

If you do not enter a value for the attribute “Statify page” in the Metadata dialog box, no statification is performed.

Example

In this example, the object “JSP page” is assigned a template cascade consisting of “JSP template” and “master JSP template”. The following table describes four sample configurations for the value of the special attribute “Statify page” and the respective results.

Table 33 – Sample configuration for the special attribute “Statify page”

JSP page	JSP template	Master JSP template	Result
`on`			The code of all three JSP objects is converted into static code. A complete statification is performed.
	`on`		The code of the JSP template and the master JSP template is converted into static code. The code of the JSP page, however, is not statified.
	`local`		The code of the JSP template is converted into static code. The code of the master JSP template and the JSP page is, however, not statified. A local statification is performed. **Note:** The head element is also statified in this case.

JSP page	JSP template	Master JSP template	Result
`on`	`local`		The code of all three JSP objects is converted into static code. The value `on` for the JSP page overwrites the value `local` for the JSP template. Thus, the master JSP template is also statified.

Variables defined in a JSP template can only be used by objects that have the same “statification status” as the JSP template. For the object to be able to use the variable from the JSP template, one of following conditions must be met in the case of a not completely statified template cascade:

- the code of both the JSP template and the object must be statified
- neither the code of the JSP template nor the code of the object may be statified.

Statification of Form Instances

Statification of form instances is always object-dependent. If, in the Admin client, statification was set to “always” for a deployment system, it is not possible to add, edit, or statify form instances by means of a JSP page in this deployment system. In a deployment system with such a configuration, you cannot work with forms in a useful way.

Although the special attribute “Statify page” is also analyzed for the statification of form instances, a different algorithm is used than with JSP objects: form instances are statified by means of a JSP page that must be specified. The special attribute “Statify page” contains the OID of the JSP page that takes over the actual statification. If this special attribute does not have a value in a form instance, the system checks the instance’s form template to see if such a value exists. If this is the case, the value is used for statification. The OID of the form instance is sent to the determined JSP page for statification. Afterwards, the result (usually an HTML page)

is generated by the deployment system. References from other objects to the form instance are automatically mapped to this HTML page.

If the *Analyze statified pages* check box is selected for the deployment system, all form instances to which this JSP page is assigned are regenerated and statified each time the JSP page changes.

The JSP page determines the layout and presentation of the data from the form instance. If no OID is stored in the special attribute “Statify page”, the deployment system generates the unchanged content of the form instance in the form of XML code.

The special attribute “Absolute URL”

When a deployment system is added in the Admin client, you determine how the references in the WCM objects are converted into URLs. The following options exist:

- *absolute:* Absolute paths are used in the links.

  Example: <a href=”http://www.company.example/products/NewProducts.htm”>

- *relative:* Relative paths are used in the links.

  Example: <a href=”NewProducts.htm”>

- *server-relative:* Relative references without specification of protocol and base URL are used in the links.

  Example: <a href=”/products/NewProducts.htm”>

In the Metadata dialog box, you can overwrite the setting of the Admin client by means of the special attribute “Absolute URL”. The following table describes several sample configurations for the setting in the Admin client and the value of the special attribute “Absolute URL”.

Table 34 – Sample configuration for the special attribute “Absolute URL”

Option in the Admin client	Possible value of the special attribute “Absolute URL”	Converted references
absolute	not set or attribute does not exist	absolute
	SERVER_RELATIVE	
	any string (e.g. on)	
relative	not set or attribute does not exist	relative
	SERVER_RELATIVE	server-relative
	any string (e.g. on)	absolute
server-relative	not set or attribute does not exist	server-relative
	SERVER_RELATIVE	server-relative
	any string (e.g. on)	absolute

Deployment systems created with the option *relative* calculate the references that the JSP page contains relative to the path of the JSP page. When the JSP page is executed for statifying a form instance, the relative URLs are written to the static code of the form instance, where they are no longer adapted. This causes problems if the form instance is not located in the same directory as the JSP page used for statification. In this case, the relative URLs are wrong. To avoid this problem, set the attribute “Absolute URL” for the JSP page to on (or to any value other than SERVER_RELATIVE). This generates absolute URLs on the executed JSP page. When the form instance is statified, these URLs are converted into relative URLs – relative to the newly generated form instance.

Statification of XML Objects

XML objects can be statified. Two cases are differentiated for statifying XML objects:

- The XML object is assigned a form template **or** the special attribute “Statify page” of the XML object has a purely numeric value.

  Statification is performed in the same way as for form instances, i.e. by means of a JSP page (see section “Statification of Form Instances” on page 418). For each XML object, the following files are generated:

  - the unstatified page
  - the statified page
  - if required: the surrogate page with a link to the XML document

  > **Note:** When a form template is used, the content of the unstatified page is distributed as HTML code (file extension **.html**) because the form template is taken into account. This means that the JSP page responsible for statification cannot directly evaluate the XML code of the XML object. For this reason, it is not recommendable to assign a form instance to an XML object.

- The special attribute “Statify page” of the XML object does not have a numeric value. No form template is assigned to the XML object. At least one of the templates assigned to the XML object is a JSP template. In this case, the template directly assigned to the XML object should be an XSLT template. For more information on using XML and XSLT, refer to appendix A “Livelink WCM Server and XML”.

Statification is performed in the same way as for JSP objects (see section “Statification of JSP Objects” on page 416). For each XML object, the following files are generated:

- the statified page
- if required: the surrogate page with a link to the XML document

Activating Thumbnails for Objects

The special attribute “Thumbnail” was introduced to enable thumbnails for objects. By default, the attribute “Thumbnails” is assigned to the attribute sets “html” and “dynamic”. These attribute sets are linked with certain object types. You can change the assignment of attribute sets to object types via *Tools* → *Website administration* → *Object types*.

Thumbnails can be displayed in the following dialogs boxes:

- When adding an object on the basis of a template, thumbnails facilitate the selection of an appropriate template.
- When selecting objects via a filter, thumbnails enable you to quickly find the desired object. A selection of objects with thumbnails is only possible for object references integrated into the generated page by means of WCM tags. Additionally, the parameters `filter` and `thumbnails` must be set for these WCM tags. For more information, refer to sections “The `filter` Parameter” on page 451 and “The `thumbnails` Parameter” on page 452. For an application example, refer to section “Content Assembly with InSite Editing” on page 331.

To activate the thumbnail for an object:

> **Note:** For graphic objects, the steps described in the following are not required. In this case, thumbnails are automatically activated.

Step 1 – Create screenshot for object

1. Navigate to the object for which a thumbnail is to be displayed.
2. Display the object in the object preview and create a screenshot for the object.
3. Save the screenshot on your local drive.

> **Note:** For templates, we recommend you draw a rough sketch of the page layout by means of a graphic program instead of the screenshot.

Step 2 – Save screenshot in the WCM system

Add an object on the basis of the graphic file created in step 1.

Step 3 – Set attribute "Thumbnail"

1. Change to the Edit view of the website.
2. Navigate to the object for which a thumbnail is to be displayed.
3. Choose *View* → *Metadata* → *Attributes* tab.
4. In the *Thumbnail* field, enter the OID of the object added in step 2.

   If you do not know the OID, click the icon ... .

   A window with the topic structure opens.

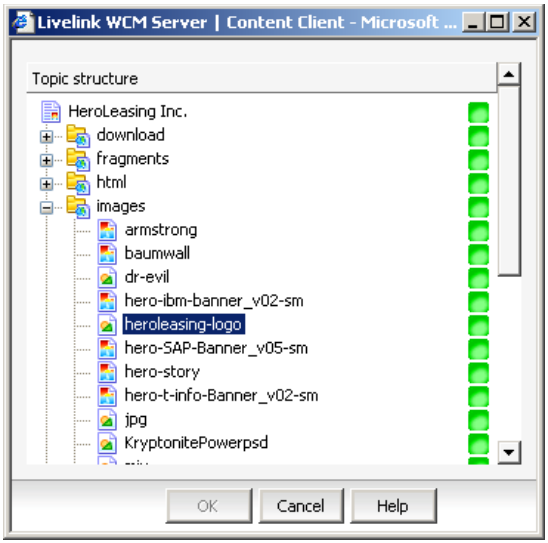


Fig. 128 – Selecting an image for a thumbnail

5. Select the desired object.

   Click the icon to view the selected object in the object preview. The icon takes you directly to the selected object.

6. Click the *Save* button.

   The thumbnail is now activated for the current object.

CHAPTER 15
Working with WCM Tags

By means of so called *WCM tags*, you can integrate metadata in a template or in an object's HTML page. WCM tags are placeholders for metadata which are inserted in the HTML code. When the pages are generated, the relevant metadata items are automatically copied into the page. In the HTML code, WCM tags are enclosed in curly brackets.

Notes:

Do not confuse WCM tags with the placeholders for certain elements that are also enclosed in curly brackets in this document (see section "Typographic Conventions" on page 23).

An overview of all information integrated in the current object by means of WCM tags is available in the Metadata dialog box on the *WCM tags tab*.

By using WCM tags in templates, you ensure that important object data is always displayed on your website in the same form. This object data includes, for example, the expiration date or the e-mail address of the editor responsible for maintaining the object. The use of WCM tags is not confined to templates and offers individual design opportunities for other objects as well. Especially in connection with the Content Assembly functions and InSite Editing, WCM tags play an important part.

This chapter explains how to

- integrate metadata into a page by means of WCM tags, see the following section
- influence the functionality of WCM tags by means of parameters, see section “Parameterized WCM Tags” on page 440

Integrating Metadata

The integrated HTML editor provides a wizard for inserting WCM tags in a comfortable way (see section “Inserting a WCM Tag in an HTML Object” on page 179).

In the following sections, the most important WCM tags are described.

Integrating object content in a page

Like any HTML page, a template may contain any desired elements (e.g. graphics, text). Since a template provides the framework for other WCM objects, it also contains the information as to where the content of the WCM object is to be inserted. For this purpose, the special WCM tag `{VIPCONTENT}` is provided.

Integrating object-specific information in a page

The following table lists the WCM tags for integrating object-specific information. The majority of this information is managed in the Metadata dialog box.

Table 35 – WCM tags for object-specific information

Metadata item	WCM tag
Author	{VIPCREATED_BY}
Country code of the language (e.g. “US” for the United States)	{VIPCOUNTRY}
Created on	{VIPDATE_CREATED}
Delayed release	{VIPDATE_RELEASED_AT}
Description	{VIPDESCRIPTION}
Direct release	{VIPDIRECT_RELEASE}
E-mail Edit, E-mail QA, E-mail Release	{VIPEMAIL_EDIT} {VIPEMAIL_QA} {VIPEMAIL_RELEASE}
Expiration date	{VIPDATE_EXPIRE}
Hierarchy level	{VIPLEVEL}
File extension	{VIPSUFFIX}
Heading	{VIPSUBTITLE}
Keywords	{VIPKEYWORDLIST}
Language	{VIPLANGUAGE}
Language code of the language (e.g. “en” for English)	{VIPDCLANGUAGE}
Modification date	{VIPDATE_MODIFIED}
Modified by	{VIPMODIFIED_BY}
Object category	{VIPOBJECTCATEGORY}

Metadata item	WCM tag
Object size (in bytes)	{VIPCONTENT_SIZE}
OID	{VIPOID}
Parent topic	{VIPTOPIC}
References from other objects to this object	{VIPLINKED_FROM} **Note:** An object containing this WCM tag is not automatically regenerated if a reference to this object is added to another object. The page for the object is generated again when the object is submitted. Alternately, you can manually start page generation (function “Regenerate page” in the Content client).
References of the object to other objects	{VIPLINKS_TO}
Release date	{VIPDATE_RELEASED}
Released by	{VIPRELEASED_BY}
Status	{VIPSTATE}
Suggested file name	{VIPDEPLOYMENT_HINT}
Target group	{VIPTARGET_GROUP}
Template	{VIPTEMPLATE}
Title	{VIPTITLE}
Type	{VIPTYPE}

Metadata item	WCM tag
URL	{VIPURL} **Note:** Deployment systems can be configured to create relative references. If the deployment system that generated the page has been configured accordingly, this URL is also relative.
URL of the statified page	{VIP_STATIC_URL} **Note:** The metadata item “URL of the statified page” only has a value in the case of form-based statification (see section “Statification of Form Instances” on page 418).
URL of surrogate page	{VIPSURROGATE_URL}
Version	{VIPVERSION}

Notes:

Each metadata item is assigned a certain data type that has a special format. This data type controls the format that the object information integrated by means of WCM tags has in the generated page. For an overview of the data types of the standard metadata and the formats used by Livelink WCM Server, refer to table “Comparison values in the filter editor” on page 388.

For information on integrating special attributes, refer to section “Integrating special attributes in a page” on page 433.

Integrating global information in a page

The following table lists the WCM tags for integrating global information.

Table 36 – WCM tags for global information

Information	WCM tag
Installation directory of the server	{VIPSERVERPATH}
VIPP port	{VIPINTRANET_PORT}
HTTP port	{VIPHTTP_PORT}
Host name of the server	{VIPHOST}
Website containing the object	{VIPSITE}, {VIPWEBSITE}
Name of the server on which the deployment system is running	{VIPSERVER}
Name of the deployment system	{VIPDEPLOYMENT_NAME}
Type of the deployment system	{VIPDEPLOYMENT_TYPE}
URL of the deployment system	{VIPDEPLOYMENT_URL}
Path to the directory where the pages generated by the deployment system are stored	{VIPDEPLOYMENT_PATH}
Host name of the mail server	{VIPMAIL_HOST}
User ID of the e-mail sender	{VIPMAIL_SENDER}

Integrating workflow information in a page

The following table lists the WCM tags for integrating workflow information.

Table 37 – WCM tags for workflow information

Information	WCM tag
Name of the workflow	{VIPWORKFLOW}
Name of the current activity	{VIPACTIVITY}

Integrating special attributes in a page

Special attributes – like all object metadata – can be integrated in pages or templates by means of WCM tags. You can form WCM tags for special attributes by adding the prefix “VIP” to the attribute and enclosing the expression in curly brackets.

The following table contains the attributes of the attribute sets supplied with Livelink WCM Server and the WCM tags formed accordingly. For displaying the attributes in the Content client, names that are easier to understand were chosen. These names are listed in the column “Attribute name in the Content client”.

Table 38 – WCM tags for the supplied special attributes

Attribute	WCM tag	Attribute name in the Content client	Data type
absolute_urls	{VIPABSOLUTE_ URLS}	Absolute URL See section “Statification” on page 413	String
applet	{VIPAPPLET}	Applet	String
att_version	{VIPATT_ VERSION}	Revision	String
bits_per_pixel	{VIPBITS_PER_ PIXEL}	Bits per pixel	Large integer

Attribute	WCM tag	Attribute name in the Content client	Data type
code	{VIPCODE}	Code	String
converter	{VIPCONVERTER}	Converters	String
copyright	{VIPCOPYRIGHT}	Copyright	String
editor	{VIPEDITOR}	Editor	String
generate_static	{VIPGENERATE_ STATIC}	Statify page See section “Statification” on page 413	String
height	{VIPHEIGHT}	Height	Large integer
object_size	{VIPOBJECT_ SIZE}	Object size	Large integer
resolution	{VIPRESOLUTION}	Resolution	Large integer
robots	{VIPROBOTS}	Robots	String
source	{VIPSOURCE}	Source	String
thumbnail	{VIPTHUMBNAIL}	Thumbnail See section “Activating Thumbnails for Objects” on page 422	OID

Attribute	WCM tag	Attribute name in the Content client	Data type
timeout	{VIPTIMEOUT}	Timeout See section “Statification” on page 413	Large integer
width	{VIPWIDTH}	Width	Large integer

In the column “Data type”, you can see which data type has been assigned to the attribute. This data type controls the format that the object information integrated by means of WCM tags has in the generated page. Moreover, the data type determines the layout of the input element for the special attribute in the Content client. The following table lists the different data types and the formats used by Livelink WCM Server and the respective input elements.

Note: The table contains the default formats used. By specifying parameters in the WCM tags, you can change the format of the object information displayed. For more information, see section “Parameterized WCM Tags” on page 440.

Table 39 – Data types with formats used

Data type	Format used	Input element
Boolean value	Returns – depending on the language setting of the WCM object – `true` or `false` or `wahr` or `falsch`.	Check box
Date	Returns – depending on the language setting of the WCM object – the date in the following form: 04.09.2003 (German) Sep 4, 2003 (English)	Separate dialog box (calendar)
External URL	Returns the entry itself	Simple text input field
Floating-point number	Returns the floating-point value. The number of digits after the decimal point (scale) is determined when the special attribute is created. The value range is defined as follows: from $-(10^{p-s}-10^{-s})$ to $+(10^{p-s}-10^{-s})$, with p = precision and s = scale Example: A precision of 5 with a scale of 3 results in a value range of –99.999 to +99.999.	Simple text input field
Formatted text	Returns the HTML content of the attribute. Only the content of the body element is returned. If there is no body element, the entire text is generated to the page. Links contained in the text are not managed automatically.	Integrated HTML editor

Data type	Format used	Input element
Integer	Returns the integer itself The value range is defined as follows: –2147483647 <= integer <= 2147483647	Simple text input field
Large integer	Returns the large integer itself The value range is defined as follows: –9223372036854775808 <= large integer <= 9223372036854775807	Simple text input field
List	Returns a comma-separated list of the entries in the order they were entered. In the case of expressions enclosed in quotes, quotes are not generated to the page.	Simple text input field
Livelink ID	Returns the ID of the object in the Livelink system	Separate dialog box for selecting a Livelink system and a Livelink object
Set	Returns an alphabetically sorted, comma-separated list of the entries. Double entries are removed. In the case of expressions enclosed in quotes, quotes are not generated to the page.	Simple text input field

Data type	Format used	Input element
Set of e-mail addresses;	Returns an alphabetically sorted, comma-separated list of the entries. Double entries are removed. In the case of expressions enclosed in quotes, quotes are not generated to the page.	Simple text input field
Object content	Returns the generated content of the referenced object. Only the content of the body element is returned. WCM tags are replaced by the metadata of the **referenced** object.	Simple input element with additional icons (e.g. for opening the topic structure)
Object source	Returns the generated content of the referenced object. Only the content of the body element is returned. WCM tags are replaced by the metadata of the **current** object.	Simple input element with additional icons (e.g. for opening the topic structure)
Object URL	Returns – depending on the settings of the deployment system – the URL of the referenced object as a relative, absolute, or server-relative reference.	Simple input element with additional icons (e.g. for opening the topic structure)
OID	Returns the OID as a string	Simple input element with additional icons (e.g. for opening the topic structure)
String	Returns the entry itself	Simple text input field

Data type	Format used	Input element
Text	Returns the entry itself	Multiline text input field
Time	Returns – depending on the language setting of the WCM object – the time in the following form: 15:59:23 (German) 3:59:23 PM (English)	Simple text input field
Time stamp	Returns – depending on the language setting of the WCM object – the date and time in the following form: 04.09.2003 15:59:23 (German) Sep 4, 2003 3:59:23 PM (English)	Separate dialog box (calendar). The time must be entered manually in the text input field.

Note: When integrating WCM objects by means of special attributes of type “Object content” or “Object source”, it may happen that a user sees the content of an object without having read rights for the object. Example: The content of object A is integrated in object B by means of a WCM tag. The user only has read rights for object B. However, via the respective page, the user can also see the content of object A.

Example

The following HTML code provides an example of the use of WCM tags.

```
<html>
      <head><title>{VIPTITLE}</title></head>
      <body>
         <table height=100% width=100%>
            <tr>
               <td height=80%>
                  <h2>{VIPSUBTITLE}</h2>
                  {VIPCONTENT}
               </td>
            </tr>
            <tr>
               <td height=20% align=right>
                  Valid until: {VIPDATE_EXPIRE}<br>
                  Contact:
                  <a href="mailto:{VIPEMAIL_EDIT}">{VIPEMAIL_EDIT}
                  </a>
               </td>
            </tr>
         </table>
      </body>
</html>
```

For detailed application examples, refer to chapters 9 “InSite Editing” and 10 “Working with Assembled Objects”.

Parameterized WCM Tags

WCM tags with parameters are called *parameterized WCM tags* and are formed according to the following pattern:

```
{<VIPTAG> <parameter1>="<value1>" ... <parameterN>="<valueN>"}
```

You can enclose the parameter value in double or single quotes. If the value does not contain any spaces, the quotes are not required.

Parameters in WCM tags fulfill multiple tasks. Parameters are provided for the following purposes:

- changing the formatting of the integrating object information on the generated page, see the following section
- enabling functions, see section “Enabling Functions” on page 451
- referencing an object, see section “Referencing Objects” on page 457
- for InSite Editing, see section “Parameters for InSite Editing” on page 458
- for Content Assembly, see section “Parameters for Content Assembly” on page 461

Formatting the Integrated Information

In order to determine how and in which language the object information integrated in the generated page by means of the WCM tag is displayed, you can use the following parameters:

- `format`, see the following section
- `locale`, see section “The `locale` Parameter” on page 449
- `encoding`, see section “The `encoding` Parameter” on page 450
- `type`, see section “The `type` Parameter” on page 450

The `format` Parameter

The `format` parameter enables you to format object information. The interpretation of the parameter depends on the data type. In the following, the possible values of the `format` parameter will be introduced for the supported data types.

Values of the `format` parameter for dates

Note: The following description is limited to the most important values. Detailed information on the formatting options for the data types “Date”, “Time”, and “Time stamp” is available at http://java.sun.com/j2se/1.4.2/docs/api/java/text/SimpleDateFormat.html.

Table 40 – Values of the `format` parameter for the data types “Date”, “Time”, and “Time stamp”

Value	Explanation	Presentation
`D`	Day in year	Number
`d`	Day in month	Number
`E`	Day in week	Text **Example:** The week day is Thursday. In this case `E`, `EE`, and `EEE` return “Thu”. `EEEE` returns “Thursday”.
`M`	Month	Number or text (depending on the number of letters) **Example:** The month is September. In this case, `M` returns “9”, `MM` returns “09”, `MMM` returns “Sep”, and `MMMM` returns “September”.
`W`	Week in month	Number
`w`	Week in year	Number
`y`	Year	Number **Example:** The year is 2003. In this case, `y`, `yy`, and `yyy` return “03”. `yyyy` returns “2003”.
`H`	Hour (0-23)	Number
`k`	Hour (1-24)	Number

Value	Explanation	Presentation
`m`	Minute	Number
`s`	Second	Number
`z`	Time zone	Text
`long`	Date Time Time stamp	Number and text **Examples** September 20, 2004 2:07:06 PM CEST September 20, 2004 02:07:06 PM CEST
`medium`	Date Time Time stamp	Number **Examples** Sep 20, 2004 2:07:06 PM Sep 20, 2004 2:07:06 PM
`short`	Date Time Time stamp	Number **Examples** 9/20/04 2:07 PM 9/20/04 2:07 PM

The following table provides examples for using the `format` parameter with dates. The examples refer to the standard metadata item “Release date” of the data type “Time stamp”, which has the value “9/20/04 2:07:06 PM”. The language of the WCM object is English.

Table 41 – Examples of the `format` parameter for the data type “Time stamp”.

Parameterized WCM tag	Output
`{VIPDATE_RELEASED format= "E, MMMM d, yyyy"}`	Thu, September 20, 2004
`{VIPDATE_RELEASED format= "HH:mm"}`	02:07
`{VIPDATE_RELEASED format= "yyyy-MM-dd 'at' HH:mm:ss z"}`	2004-09-20 at 02:07:06 PM CEST

Values of the `format` Parameter for Object URLs

By default, the format of the URL returned for the data type “object URL” is determined by the corresponding setting of the deployment system. The `format` parameter enables you to control the format. In this case, the WCM system ignores the deployment system setting and the value of the special attribute “absolute_url”.

The following values are possible for the `format` parameter:

- `relative`: Relative paths are used in the URLs.
- `absolute`: Absolute paths are used in the URLs.
- `server-relative`: Relative references without specification of protocol and base URL are used in the URLs.
- `context-relative`: Relative references without specification of protocol, base URL, and web application name are used in the URLs. This setting can only be used if an application name has been configured for the deployment system.

The following table provides examples for using the `format` parameter with the data type “Object URL”.

Table 42 – Examples of the `format` parameter for the data type "Object URL"

Parameterized WCM tag	Output
`{VIPURL format= "relative"}`	NewProducts.htm
`{VIPURL format= "absolute"}`	http://wcmserver.company.example/wcm/products/NewProducts.htm
`{VIPURL format= "server-relative"}`	/wcm/products/NewProducts.htm
`{VIPURL format= "context-relative"}`	/products/NewProducts.htm

Values of the `format` parameter for floating-point numbers

Note: The following description is limited to the most important values. For detailed information on the format description and the formatting options for the data type "Floating-point number", refer to http://java.sun.com/j2se/1.4.2/docs/api/java/text/DecimalFormat.html and http://java.sun.com/docs/books/tutorial/i18n/format/decimalFormat.html.

Table 43 – Values of the `format` parameter for the data type "Floating-point number".

Value	Explanation
0	Digit, if the digit does not exist, "0" is returned
#	Digit
-	Minus sign
, (comma)	Grouping separator
. (dot)	Decimal separator

Value	Explanation
E	Separates mantissa and exponent
percent	Multiplies by 100 and shows the value in percent
\u2030	Multiplies by 1000 and shows the value per mill
\u00A4	Returns the currency symbol, depending on the language of the WCM object or the value of the locale parameter
\u00A4\u00A4	Returns the international currency abbreviation, depending on the language of the WCM object or the value of the locale parameter
currency	Inserts separators for thousands and decimal places; returns the currency symbol, depending on the language of the WCM object or the value of the locale parameter

The following table provides examples for using the format parameter with the data type “Floating-point number”. The examples refer to a WCM tag for a special attribute of type “Floating-point number” and a scale of 4. The language of the WCM object is English.

Table 44 – Examples of the format parameter for the data type “Floating-point number”

Value of the floating point-number	Value of the format parameter	Output
1234567.9999	not set	1,234,567.9999
1234567.1234	#,###,###.## \u00A4	1,234,567.12 $
234567.9999	0,000,000.00 \u00A4\u00A4	0,234,568.00 USD
234567.9999	0.####E00	2,3457E05

Value of the floating point-number	Value of the `format` parameter	Output
0.87	`currency`	0.87 $
0.25	`percent`	25 %
0.005	`### \u2030`	5 ‰

Values of the `format` parameter for integers and large integers

Possible values for the `format` parameter are:

- `currency`: Inserts separators for thousands and decimal places; returns the currency symbol, depending on the language of the WCM object or the value of the `locale` parameter

  Examples:

  The value 42 is output as 42.00 $.
  The value 123456789 is output as 123,456,789.00 $.

- `bytes`: returns the unit bytes, KB, or MB, depending on the amount of the value

  Examples:

  The value 42 is output as 42 bytes.
  The value 1234 is output as 1.20 KB.
  The value 123456789 is output as 117.73 MB.

Values of the `format` parameter for Livelink IDs

Possible values for the `format` parameter are:

- `nodeid`: returns the ID of the object in the Livelink system
- `systemname`: returns the name of the Livelink system

Values of the `format` parameter for sets and lists

By default, commas are interpreted as separators for the data types “Set”, “Set of e-mail addresses”, and “List”. You can use the `format` parameter to specify a different separator. For example, `{VIPKEYWORDLIST}` returns “Vermont, Maine, Connecticut”, whereas `{VIPKEYWORDLIST format=" | "}` returns “Vermont | Maine | Connecticut”.

You can also use WCM tags to integrate attributes from a Livelink category (see section“Integrating Livelink Metadata in a Page” on page 358). As Livelink attributes may have more than one value, all Livelink attributes have set types. Use the `format` parameter to format integrated object information in the usual way (e.g. in order to determine how time stamps are displayed). In order to **simultaneously** specify a separator for multiple attribute values, use the `separator` parameter.

Example:

You have a Livelink category name “Training” which, among others, contains the attribute “Dates”. The dates of the next three trainings are saved as value of the “Dates” attribute. The WCM tag

```
{VIPLL category="Training" attribute="Dates" format="&amp;nbsp;|&amp;nbsp;"}
```

outputs “Mon Nov 29 00:00:00 CET 2004 | Mon Jan 10 00:00:00 CET 2005 | Mon Mar 07 00:00:00 CET 2005”.

To format the dates in a more attractive way, use the WCM tag

```
{VIPLL category="Training" attribute="Dates" format="yyyy-MM-dd"
separator="&amp;nbsp;|&amp;nbsp;"}
```

This WCM tag outputs “2004-11-29 | 2005-01-10 | 2005-03-07”.

The `locale` Parameter

By means of the `locale` parameter, you can localize the information integrated via WCM tags. You can set the language version for displaying the information and the country-specific conventions for formatting the data. If the `locale` parameter is not set, the system uses the WCM object's language setting.

The values of the `locale` parameter are specified in the form `language code_country code` according to ISO-639 and ISO-3166 (e.g. `en_US` for English). A list of the language codes according to ISO-639 is available at http://www.w3.org/WAI/ER/IG/ert/iso639.htm. For an overview of the country codes according to ISO-3166, refer to http://www.iso.org/iso/en/prods-services/iso3166ma/02iso-3166-code-lists/list-en1.html.

You can use the `locale` parameter for the following data types:

- **Boolean value:** returns “wahr” or “falsch” for the value `de_DE`. All other values return “true” or “false”.
- **Set, Set of e-mail addresses:** determines the sort order
- **Date, Time, Time stamp:** returns the date, time, the date with time in a country-specific format

  Example: `{VIPDATE locale="en_US"}` returns “Sep 4, 2003”; `{VIPDATE locale="de_DE"}` returns “04.09.2003”.
- **Floating-point number:** returns the floating-point number in country-specific format

Note: For programming JSP pages, it may be required to output the internal names of object statuses, object types, and default values. To do so, set the `locale` parameter to `none`. Example: `{VIPSTATE locale="none"}` returns “edited” for the object status “changed”.

The encoding Parameter

The encoding parameter determines the encoding to be used to write the value of a WCM tag to the page. Permitted values are:

- plain: The encoding configured for the website is used (ISO-8859-1 or UTF-8).
- html: The text is HTML-encoded. Non-ASCII characters are output as Unicode in HTML (numeric entities; example: “Ä” is replaced by “Ä”).
- javascript: The text is coded in order to enable the use in Javascript code. The following characters are replaced:

" (one character)	→	\" (two characters)
' (one character)	→	\' (two characters)
\ (one character)	→	\\ (two characters)
\r (one character)	→	\r (two characters)
\n (one character)	→	\n (two characters)
\t (one character)	→	\t (two characters)
<	→	<'+'

Notes:

Values that return URLs are not encoded.

If the parameter encoding is not specified, HTML encoding is always used. In this case, the named entity is used (example: “Ä” is replaced by “Ä”).

The type Parameter

Use the type parameter to specify the HTML tag generated for the data type “Object URL”. Possible values of the parameter are img and href. In

the generated page, the title of the referenced object is used for the link text.

Table 45 – Examples for using the `type` parameter

Parameterized WCM tag	Generated URL
`{VIPNEWS_IMAGE type="img"}`	`<img src="..." title="{title of the referenced object}" />`
`{VIPNEWS_DATASHEET type="href"}`	`<a href="...">title of the referenced object</a>`

Enabling Functions

The following parameters are provided for enabling special functions:

- `filter`, see the following section
- `thumbnails`, see section “The `thumbnails` Parameter” on page 452
- `default`, see section “The `default` Parameter” on page 452
- `required`, see section “The `required` Parameter” on page 453
- `label`, see section “The `label` Parameter” on page 453
- `option_id` and `option_mode`, see section “The `option_id` and `option_mode` Parameters” on page 454

The `filter` Parameter

For the data types “Object content”, “Object source”, “Object URL”, and “OID”, which represent references to other WCM objects, you can use the `filter` parameter to limit the number of objects to be selected. The syntax of the filter expression corresponds to the syntax used by the object filter in the Content client.

Note: The value of the `filter` parameter must **always** be enclosed in quotes (example: `filter="(topic=5)"`). Otherwise, problems may occur when the filter expression is evaluated. If the filter expression already contains double quotes (example: `oty_name="GIF"`), enclose the `filter` parameter value in single quotes (example: `filter='(oty_name="GIF")'`).

For further information on creating filters, refer to section “The Filter Editor” on page 376. For an example of using a filter in WCM tags, refer to section “Content Assembly with InSite Editing” on page 331.

The `thumbnails` Parameter

If the `thumbnails` parameter has the value `on`, thumbnails of the individual objects are displayed when selecting objects by means of a filter. For this reason, you should use the `thumbnails` parameter only in connection with the `filter` parameter. In the case of graphic objects, the thumbnails are displayed automatically. For all other WCM objects, the special attribute “Thumbnail” must be set. For more information, refer to section “Activating Thumbnails for Objects” on page 422. For an example of using the `thumbnails` parameter, refer to section “Content Assembly with InSite Editing” on page 331.

The `default` Parameter

By means of the `default` parameter, you can use values from the template in the generated page of the object. This presupposes that the value of the parameter is `on` and that the respective metadata item is not set in the WCM object. In the Metadata dialog box, the value from the template is not displayed. If a cascading template is assigned to the object, the system searches the template cascade until it finds a respective value.

Notes:

The `default` parameter can be used for all special attributes and for the following standard metadata: “Heading”, “Direct release”, “Keywords”, “Description”, “Target group”, “E-mail Edit”, “E-mail QA”, “E-mail Release”.

If the reference to a metadata item is distributed over several WCM tags, the `default` parameter must be placed in the tag that adopts the value from the template.
Example: `{VIPIMAGEURL edit="begin"}<img src="{VIPIMAGEURL default="on"}">{VIPIMAGEURL edit="end"}`.

Example

The WCM object “Partner” does not have an entry for the “Partnername” property. The template contains the entry “Enter the name of the partner” for this property. `{VIPPARTNERNAME default="on"}` writes the entry from the template to the generated page of the WCM object “Partner”. This way, the user editing the WCM object knows immediately what to enter into this field.

The `required` Parameter

Use the `required` parameter to determine that the field of the corresponding metadata item in the WCM tag dialog box must be filled in (example: {VIPSUBTITLE required=true}). If the parameter is not set, the WCM system does not check whether the field is filled in.

The `label` Parameter

By means of the `label` parameter, you can define alternative names for special attributes. The alternative names are displayed for the respective object in the Metadata dialog boxes of the Content client and of InSite Editing.

If the Content client is used in different languages, you can localize, i.e. translate, the alternative names. In this way, the translated alternative names of the special attributes will be shown in a German user interface.

To localize an alternative name, add the value of the `label` parameter to the file **FieldNameMessage_*.properties**. The asterisk * replaces the language label of the file, e.g. “en” for English and “de” for German. For detailed information on localizing user interface texts, refer to the Livelink WCM Server Administrator Manual (chapter “Working in the Admin Client”).

Example

Your object category “Press_Release” contains a number of properties, including “main”. The WCM tag `{VIPMAIN label=”Main text”}` causes the special attribute “main” to be shown with the name “Main text” in the Metadata dialog box.

In order to provide a translated alternative name in a German user interface, add the following entry to the file **FieldNameMessage_de.properties**:

```
Main text=Haupttext
```

The `option_id` and `option_mode` Parameters

Use the `option_id` parameter to set predefined values for an attribute so that the users are able to select one of these values. When a user clicks the corresponding attribute field in the Metadata dialog box, a selection dialog box with the defined values opens. The users cannot enter free text in the attribute field. You can localize the attribute values, i.e. define separate values for different languages.

To use the option_id parameter:

1. Define the values for the attribute in an XML file based on the following DTD:

```
<!DOCTYPE optionset [
    <!ELEMENT optionset (option*) >

    <!ELEMENT option (value*) >
    <!ATTLIST option name CDATA #REQUIRED >

    <!ELEMENT value (#PCDATA) >
    <!ATTLIST value locale CDATA #REQUIRED >
]>
```

2. On the basis of this XML file, add an object of type “XML document” to the website.
3. Add the attribute's WCM tag to a website object and specify the option_id parameter for this tag. As the parameter value, enter the OID of the XML document.

   By means of the option_mode parameter, you can determine whether the users are allowed to select only one or multiple values. Permitted values for the option_mode parameter are:

   - single: The user may select exactly one value. This is the default setting if the option_mode parameter is not set.
   - multiple: Multiple values may be selected.

Example

You want to ensure that systematic keywords are entered in the Metadata dialog box in order to describe the customer invoices. In this way, you want to facilitate quick searches for the invoices in the Content client. Depending on the invoice's language, the keywords are to be in English and German, respectively.

The following code sample shows the contents of the XML document defining the attribute values. The OID of the XML document is 123.

```
<!DOCTYPE optionset [
      <!ELEMENT optionset (option*) >

      <!ELEMENT option (value*) >
      <!ATTLIST option name CDATA #REQUIRED >

      <!ELEMENT value (#PCDATA) >
      <!ATTLIST value locale CDATA #REQUIRED >
]>

<optionset>
      <option name="Accounting">
         <value locale="de_DE">Buchhaltung</value>
         <value locale="en_US">accounting</value>
      </option>
      <option name="Invoice">
         <value locale="de_DE">Rechnung</value>
         <value locale="en_US">invoice</value>
      </option>
      <option name="Company A" />
      <option name="Company B" />
      <option name="Company C" />
</optionset>
```

The WCM tag `{VIPKEYWORDLIST option_id="123"}` has the effect that users clicking the *Keywords* field in the Metadata dialog box of an English invoice see the following selection dialog box:

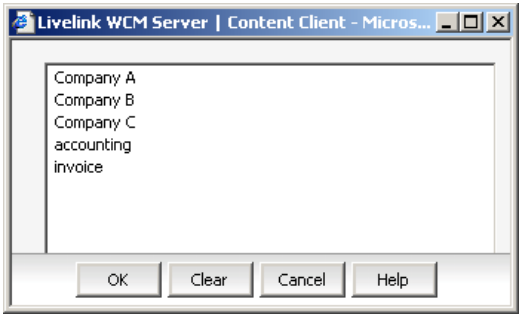


Fig. 129 – Selection dialog box for the `VIPKEYWORDLIST` attribute

In this dialog box, the users can select exactly one entry. In order to enable multiple selections, the WCM tag must look as follows:
`{VIPKEYWORDLIST option_id="123" option_mode="multiple"}`

Referencing Objects

By means of the `oid` parameter, you can integrate the metadata of another WCM object in the HTML page of the current object.

Example: `{VIPTITLE oid="7"}` generates the title of the WCM object with the OID 7.

> **Note:** If the `oid` parameter is set, it is not possible to make the value of the respective WCM tag editable in InSite Editing (see section “The `edit` Parameter” on page 458).

Parameters for InSite Editing

For working with InSite Editing, the following parameters are available:

- `edit`, see the following section
- `order`, see section “The `order` Parameter” on page 459
- `buttons`, see section “The `buttons` Parameter” on page 460
- `add`, see section “The `add` Parameter” on page 461

The `edit` Parameter

Notes:

The `edit` parameter can be used for all special attributes and for the following standard metadata: “Heading”, “Direct release”, “Delayed release”, “Keywords”, “Description”, “Target group”, “E-mail Edit”, “E-mail QA”, “E-mail Release”.

Furthermore, you can use it for the WCM tag `{VIPCONTENT}` in templates. Thus, you enable InSite Editing for the content of the objects using these templates.

By means of the `edit` parameter, you can activate InSite Editing for the information integrated into the page via WCM tags. Permitted values are:

- `on`: The information is equipped with a toolbar and highlighted by a red frame. You can edit the information directly on the page or in the WCM tag dialog box.

  Example: `{VIPKEYWORDLIST edit="on"}`

- `param`: The information can be edited via the WCM tag dialog box; they are not highlighted on the page.

  Example: `{VIPKEYWORDLIST edit="param"}`

- `begin` and `end`: In most cases, it is sufficient to enable InSite Editing by means of the value `on` or `param`. As a URL can only be used within such HTML tags as `<a href={VIPLINK} ...>` or `<img src={VIPLINK} ...>`, and these HTML tags may not contain the `edit` parameter, the parameter was extended by the values `begin` and `end`. Two additional WCM tags must be inserted that enclose the actual WCM tag. These WCM tags are only considered by InSite Editing deployment systems. All other deployment systems ignore them.

  Example: `{VIPLINK edit="begin"}<img src="{VIPLINK}"> {VIPLINK edit="end"}`

The `order` Parameter

Use the `order` parameter to determine the position of the WCM tag in the WCM tag dialog box of InSite Editing. This enables you to position the input fields for essential information on top of the dialog box. You can only enter integers for the `order` parameter.

When determining the order of tags in the WCM tag dialog box, the system evaluates the `order` parameter first. WCM tags for which this parameter is not set are sorted according to their occurrence in the object content (following the tags for which the parameter is set).

Example

The object contains the following WCM tags in this sequence:

1. `{VIPATTRIBUTE_A}`
2. `{VIPATTRIBUTE_B}`
3. `{VIPATTRIBUTE_C order="2"}`
4. `{VIPATTRIBUTE_D order="1"}`

In the WCM tag dialog box, the tags are sorted as follows:

1. Attribute_D
2. Attribute_C
3. Attribute_A
4. Attribute_B

The `buttons` Parameter

In InSite Editing, each object fragment has its own little toolbar. You can use the InSite Editing configuration file to permanently hide individual icons from this toolbar (see section “Configuring InSite Editing” on page 309). By means of the `buttons` parameter, you can display hidden icons for individual object fragments. Permitted values for the parameter are:

- `new`: displays the icon *assembled object*
- `edit`: displays the icon *Change value* or *Select different object*
- `checkout`: displays the icon *Check out*
- `checkin`: displays the icon *Check in*
- `goto`: displays the icon *Go to object*
- `param`: displays the icon *WCM tags*

To unhide multiple icons for an object fragment, enter the values separated by commas (example: `{VIPDATASHEET buttons="checkout,checkin"}`).

The add Parameter

By means of the `add` parameter, you can make additional metadata within an HTML tag, such as `<a href={VIPLINK} ...>` or `<img src={VIPLINK} ...>` editable in InSite Editing. The value of the `add` parameter corresponds to the name of the respective metadata item. The `add` parameter must be inserted directly after the `edit="begin"` parameter. In InSite Editing, an icon for editing the medadata item is displayed for each metadata item in the `add` parameter.

For example, a reference to a graphic object can be made editable within an external link as follows:

```
{VIPEXTERNAL_URL edit="begin" add="image"}<a href="
{VIPEXTERNAL_URL}"><img src="{VIPIMAGE}"/></a>
{VIPEXTERNAL_URL edit="end"}.
```

Parameters for Content Assembly

By means of the `target` parameter, you can determine whether the information of the referenced object fragment or the information of the assembled object is displayed in the generated page (when working with Content Assembly). The `target` parameter can be used for attributes with the data types “Object content” and “Object source”; permitted parameter values are `self` and `outer`. The following table illustrates the relation between data type and value of the `target` parameter.

Table 46 – Data types and `target` parameter

Data type	`target` parameter	Displayed information
Object content	not specified	of the object fragment
	Value = `self`	of the object fragment
	Value = `outer`	of the assembled object
Object source	not specified	of the assembled object
	Value = `self`	of the object fragment
	Value = `outer`	of the assembled object

CHAPTER 16

Managing Websites in the Content Client

The entire management of your WCM system is performed via the Admin client. Certain website administration functions are also available in the Content client. These include:

- adding, editing, and deleting object types, see the following section
- adding, editing, and deleting attribute sets, see section “Managing Attribute Sets” on page 477
- adding, editing, and deleting object categories, see section “Managing Object Categories” on page 485

Note: For managing websites via the Content client, you require the administration right “Website administration”. Administration rights are assigned in the Admin client. Please refer to the Livelink WCM Server Administrator Manual for additional information.

Managing Object Types

For general information on object types, refer to section “Object Types” on page 65.

Object types are managed separately for each website. After a website has been created, the default object types are available. To view the object types of the current website, choose *Tools* → *Website administration* → *Object types*. The *Object types* dialog box opens. The object types are displayed in a list in the left window pane. To view the settings for an individual object type, click it in the list.

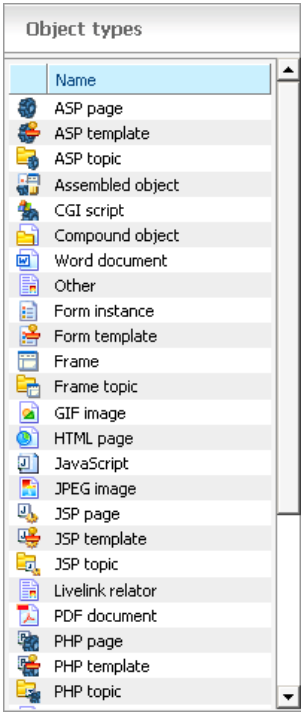


Fig. 130 – Overview of available object types

The following functions are available to you:

- adding an object type, see the following section
- editing object type settings, see section “Editing an Object Type” on page 475
- deleting an object type from the website settings, see section “Deleting an Object Type” on page 476

Adding an Object Type

In addition to the default object types, you can create custom object types.

> **Notes:**
>
> You can change the object type of WCM objects subsequently for selected object types. This is **not** possible for custom object types.
>
> In theory, you can add the object type “XML topic”. However, Livelink WCM Server does not support such an object type at present.

The following sections describe how to add an object type.

Starting the Wizard

To start the wizard:

1. Choose *Tools* → *Website administration* → *Object types*.

   The *Object types* dialog box opens.

2. Choose *New object type* on the context menu or click the corresponding icon.

   Icon for adding an object type

3. Follow the instructions of the wizard that guides you through the process.

Specifying General Information for the Object Type

In the first dialog box, you specify general information for the object type.

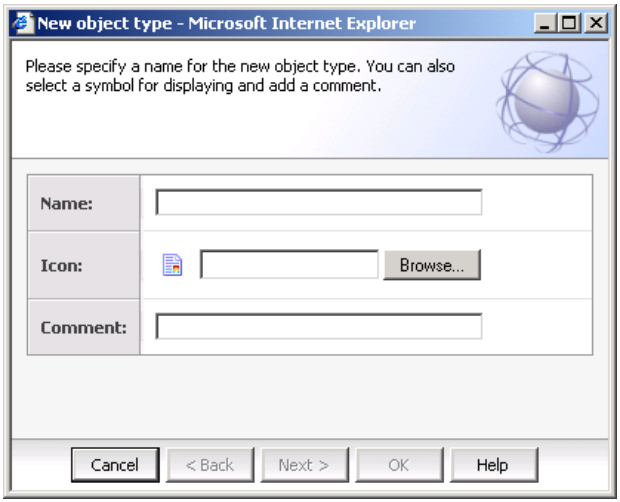


Fig. 131 – Specifying general information for the object type

- *Name:* Enter a name for the new object type. The name must be unique within a website and cannot be changed later. The name must not be more than 17 characters long. The following characters are permitted:
  - a-z and A-Z (no umlauts and special characters)
  - 0–9, the first character must always be a letter
  - _ (underline)

> **Note:** The following names are reserved and may not be used: admin, acl, ats, att, cda, con, dyn, lat, lin, obj, oty, par, pri, pro, pty, rli, sta, tta, tto.

The name of the new object type is used to generate an *internal name*. Among other things, this internal name is used to access WCM objects of a given type via the WCM Java API.

> **Note:** If you want the name of the object type to be displayed in the language of the user, you can translate it. For more information, refer to the Livelink WCM Server Administrator Manual.

- *Icon:* Select a file in GIF format as the icon for displaying the object type in the Content client. The selected graphic is copied to the following directories:
  - **{WCM installation directory}\config\vipimages\objecttype**
  - **{WCM installation directory}\admin\config\vipimages\objecttype**
  - **{WCM installation directory}\website\cmsclient\images\vipimages\objecttype**
- *Comment:* Enter any description of the object type.

Confirm your entries by clicking the *Next* button.

Specifying Object Data Type and Additional Settings

In this dialog box, you determine the object data type and specify additional settings.

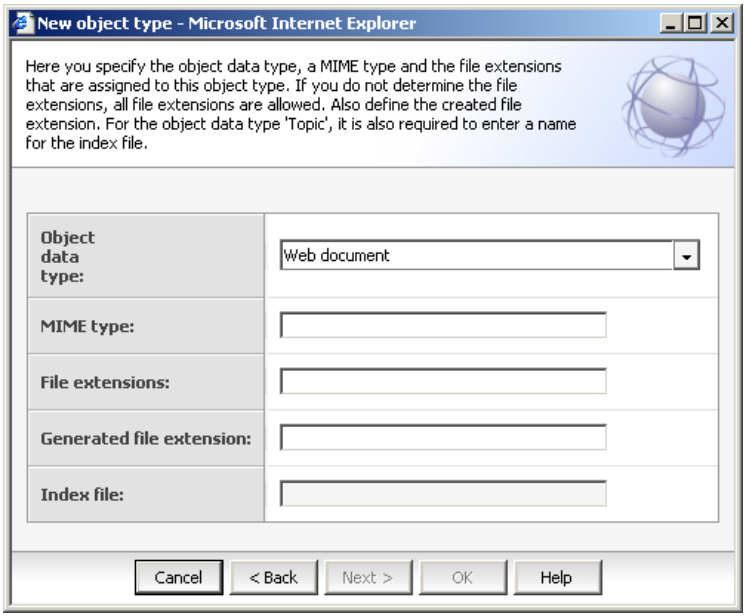

New object type - Microsoft Internet Explorer
Here you specify the object data type, a MIME type and the file extensions that are assigned to this object type. If you do not determine the file extensions, all file extensions are allowed. Also define the created file extension. For the object data type 'Topic', it is also required to enter a name for the index file.
Object data type:
Web document
MIME type:
File extensions:
Generated file extension:
Index file:
Cancel
< Back
Next >
OK
Help

Fig. 132 – Specifying object data type and additional settings for the object type

- *Object data type:* Each object type belongs to a specific object data type. The following object data types may be used for new object types.

Table 47 – Object data types and WCM objects

Object data type	WCM objects
Assembled object	Objects that do not have a content component. All data of the object is saved in the metadata. This way, the content of other objects can be integrated in the object.
Compound object	Logically related objects, e.g. HTML pages generated by means of a conversion program from a word processor document that are to be treated as a single document
Frame	Pages designed with frames – you can use this object data type, for example, to define PHP or JSP frames.
Frame topic	Objects that have subordinate frame objects
HTML template	Template for HTML documents
Livelink relator	WCM objects referencing a Livelink object
Surrogate	This object data type can be used to administer WCM objects that are not typical web documents. These include graphics, PDF documents, Microsoft Office files, etc. Hyperlinks from these documents to other objects are not managed by the automatic reference check in Livelink WCM Server. This means that the references are not analyzed when such objects are imported. However, hyperlinks from other WCM objects to objects of the object data type “Surrogate” are automatically administered by Livelink WCM Server. If an object of this type is added in the Content client, a file must be specified as content.
Topic	Objects that have child objects: Topic, ASP topic, etc.
Web document	HTML pages, JSP pages, PHP pages, JavaScript, and ASP pages

Object data type	WCM objects
Workflow	Objects in which a workflow is saved
XML document	All types of XML documents, e.g. XMath This object data type is also used for XSLT documents.
XML template	Template for XML documents
XSLT template	Template for XSLT documents

Click one of the available object data types in the drop-down list. You cannot change this entry later.

- *MIME type:* The MIME type defines how the generated pages are displayed in the browser.

  The following table shows a number of MIME types. The five basic types are supplemented by subtypes, such as “text/plain” for simple text files.

Table 48 – MIME types and WCM objects

MIME type	Sub-type and respective WCM objects
text	/plain – simple text files /asp –ASP pages /html – HTML pages /jsp – JSP pages /php – PHP pages
image	/gif – images in GIF format /jpg – images in JPG format /png – images in PNG format
audio	/basic – audio files

MIME type	Sub-type and respective WCM objects
video	/mpeg – video files in MPEG format
application	Files that do not belong to the other categories and thus have to be processed by a particular application before they can be displayed in the browser /x-javascript – JavaScript /vnd.ms-excel – Microsoft Excel /vnd.ms-powerpoint – Microsoft PowerPoint /msword – Microsoft Word /postscript – Postscript files /xml – XML files

Enter the MIME type in the text field.

- *File extensions:* extensions of the source files for importing objects, e.g. “xls” (MS Excel) or “doc” (MS Word). Enter the file extensions as a comma-separated list.
- *Generated file extension:* extension of the pages generated by the deployment system. If files cannot be displayed directly in the browser, e.g. Excel files, a surrogate page containing a link to the file is generated. In such cases, the surrogate page is generated with the file extension specified here (example: “htm”).

  In the metadata of a WCM object, a suggested file name for the generated file can be entered. When a WCM object is added or when changes are made to the object’s metadata, an automatic check is made to see whether the suggested file name’s extension is supported by the object type. Invalid file extensions cannot be saved.

  Enter the desired file extension.

- *Index file* (only for object types with the object data type “Topic”): file name of the parent topic file referencing the subordinate files. This name is used by the relevant deployment system for the generated topic file, even if a different suggestion for the file name exists in the topic’s metadata.

  Enter the respective file name.

Confirm your entries by clicking the *Next* button.

Making the Assignments of the Object Type

In this dialog box, you can assign the new object type a functional area and an attribute set.

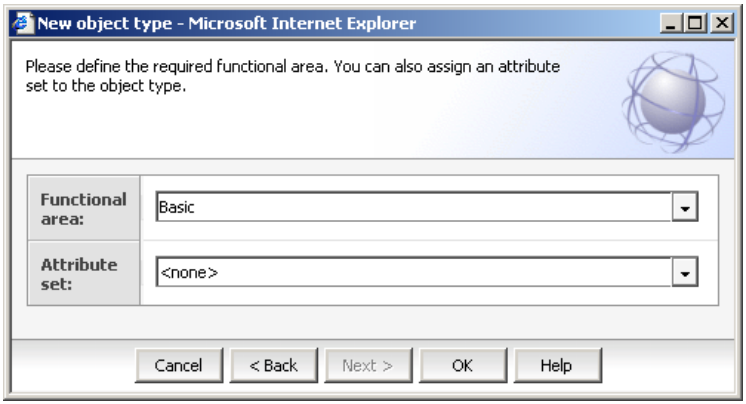

New object type - Microsoft Internet Explorer
Please define the required functional area. You can also assign an attribute set to the object type.
Functional area:
Basic
Attribute set:
<none>
Cancel
< Back
Next >
OK
Help

Fig. 133 – Making assignments for the object type

- *Functional area:* Assign a functional area to the object type. Only principals with this functional area can add, check out, and check in WCM objects of this type.

  Click one of the available functional areas in the drop-down list.

- *Attribute set:* You can assign an attribute set to the object type. The attributes of the attribute set are displayed on the *Attributes* tab of the Metadata dialog box.

  Click one of the available attribute sets in the drop-down list.

Add the object type by clicking the *OK* button.

Editing an Object Type

To change the settings of an object type:

1. Choose *Tools* → *Website administration* → *Object types*.

   The *Object types* dialog box opens.

2. Mark the desired object type in the left window pane.

   In the right window pane, three tabs with the settings for the object type are displayed. All these properties were defined when the object type was added. You can modify the following parameters subsequently.

   - *General* tab: *Icon* and *Comment*.
   - *Object data type and additional settings* tab: all entries except the object data type
   - *Assignments* tab:all entries, i.e. assignment of an attribute set and a functional area

3. Make the desired changes.
4. Click the *Save* button.

Notes on changing object types

- If you assign a different attribute set to an object type and WCM objects with this object type already exist, the attributes and values no longer assigned are removed from the metadata of the relevant WCM objects. The removed attributes and the associated values remain stored in the database as long as no changes are made to the newly assigned attribute set for the relevant WCM objects.

  Thus, if you accidently assign the wrong attribute set and then correct your mistake immediately without any changes being made to the metadata for the relevant WCM objects, the old values will be available again.

- The servers of the WCM system are automatically notified about changes to the object types, with the result that new icons are available in the Admin client, for example. However, if you installed Admin clients on computers where no WCM server exists, the changes are not available there.

Deleting an Object Type

Notes:

You can only delete an object type if no current versions of objects exist in the Edit, QA, or Production view that use this object type. Older versions of the objects may, however, exist.

If an object version is restored that uses the deleted object type, the object type will also be restored. You can use the utility “Database clean-up” to irretrievably remove a (deleted) object type. This presupposes that there are no older versions of objects using this type. For more information, refer to the Livelink WCM Server Administrator Manual.

To delete an object type:

1. Choose *Tools* → *Website administration* → *Object types*.

   The *Object types* dialog box opens.

2. Mark the desired object type in the left window pane.
3. Choose *Delete object type* on the context menu or click the corresponding icon.

   Icon for deleting an object type

4. Confirm the security prompt by clicking the *Yes* button.

Managing Attribute Sets

For general information on attribute sets, refer to section “Attributes from the attribute set” on page 409 and to the Livelink WCM Server Administrator Manual.

Some of the object types generated during creation of the website are by default assigned to attribute sets. You can change this assignment for existing object types. When adding an object type, you can optionally assign an attribute set.

To view the available attribute sets of the current website, choose *Tools* → *Website administration* → *Attribute sets*. The *Attribute sets* dialog box opens. The attribute sets are displayed in a list in the left window pane. To view the attributes of an attribute set, select the attribute set in the list.

The attribute sets for the default object types are created automatically during installation: “compound”, “dynamic”, “html”, “image”, and “other”.

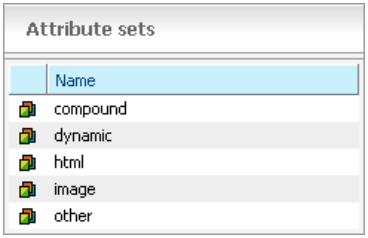


Fig. 134 – Default attribute sets

The following functions are available to you:

- adding an attribute set, see the following section
- adding attributes to an existing attribute set or removing them, see section “Editing an Attribute Set” on page 480
- deleting attribute sets from the website settings, see section “Deleting an Attribute Set” on page 484

Note: If the proxy Content servers assigned to the respective website use separate data storages, changes to attribute sets must also be saved there. For this reason, the system checks whether these proxy Content servers are available and whether the JDBC pools assigned can be used. If this is not the case, you cannot save the changes to the attribute sets. Please refer to the Livelink WCM Server Administrator Manual for additional information.

Adding an Attribute Set

To add an attribute set:

1. Choose *Tools* → *Website administration* → *Attribute sets*.

   The *Attribute sets* dialog box opens.

2. Choose *New attribute set* on the context menu or click the corresponding icon.

   Icon for adding an attribute set

   The *New attribute set* dialog box opens.

3. Enter a name for the new attribute set. The name of the attribute set must be unique within the website and must not be more than 17 characters long. The following characters are permitted:

   - a-z and A-Z (no umlauts and special characters)
   - 0–9, the first character must always be a letter
   - _ (underline)

   Note: The following names are reserved and may not be used: admin, acl, ats, att, cda, con, dyn, lat, lin, obj, oty, par, pri, pro, pty, rli, sta, tta, tto.

   You cannot change the name subsequently.

4. Add the attribute set by clicking the *OK* button.

5. Determine the attributes for the new attribute set (see the following section).

Editing an Attribute Set

Note: When attribute sets are changed, the pages of the WCM objects whose object type is assigned the attribute set are **not** automatically regenerated. In this case, it may happen that the values of attributes that were removed from the attribute set are still displayed on the generated pages.

The following functions are available for editing attribute sets:

- creating attributes, see the following section
- adding existing attributes, see section “Adding Attributes” on page 483
- removing attributes from an attribute set, see section “Removing Attributes from an Attribute Set” on page 483

Creating Attributes

Notes:

You cannot modify existing attributes.

You cannot delete attributes from the WCM system. Attributes that have been removed from an attribute set remain in the database and can be assigned again.

To create an attribute for an attribute set:

1. Choose *Tools* → *Website administration* → *Attribute sets*.

   The *Attribute sets* dialog box opens.

2. Mark the desired attribute set in the left window pane.
3. Choose *New attribute* on the context menu or click the corresponding icon.

Icon for creating an attribute

The *New attribute* dialog box opens.

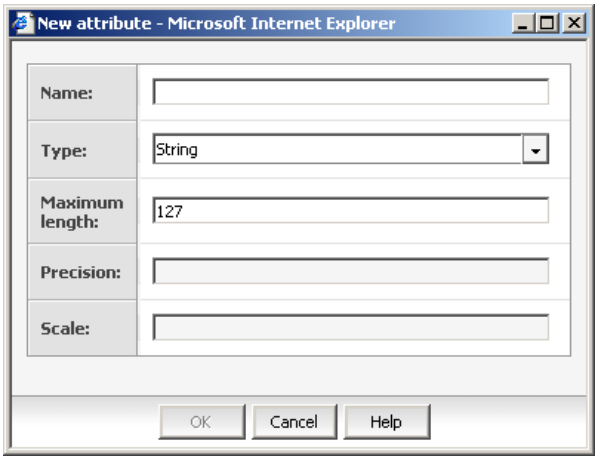


Fig. 135 – Dialog box when creating an attribute

4. Make the following settings:

   - *Name:* The name must be unique within the entire website. Moreover, the name of the attribute must not be identical to the name of a property that belongs to an object category in this website.

     The name must not be more than 20 characters long. Uppercase letters may not be used (they are automatically mapped to lower-case letters). Otherwise, the same conditions as for names of attribute sets apply.

> **Note:** If you want the name of the attribute to be displayed in the language of the user, you can translate it. For more information, refer to the Livelink WCM Server Administrator Manual.

- *Type:* Click a type in the drop-down list. For detailed information on the individual types, refer to table “Data types with formats used” on page 436.
- *Maximum length:* For the types “External URL”, “List”, “Set”, “Set of e-mail addresses”, and “String”, you must determine the maximum length. Select a high value because, for example, in the Oracle RDBMS some special characters are represented by 2 bytes instead of 1 byte. When UTF-8 is used, a character may be encoded with up to 4 bytes. The maximum length of character strings is determined by the RDBMS used (Oracle: 4000 characters).
- *Precision* (only for the type “Floating-point number”)*:* determines the number of digits to be considered
- *Scale* (only for the type “Floating-point number”): determines the number of digits after the decimal point, i.e. where the point is inserted in the numeric string

  The value range for the type “Floating-point number” is defined as follows: from $-(10^{p-s}-10^{-s})$ to $+(10^{p-s}-10^{-s})$,
  with p = precision and s = scale
  Example: A precision of 5 with a scale of 3 results in a value range of –99.999 to +99.999.

5. Click the *OK* button.

   The new attribute is added to the attribute set.

6. Click the *Apply* button.

Adding Attributes

To add existing attributes to an attribute set:

1. Choose *Tools* → *Website administration* → *Attribute sets*.

   The *Attribute sets* dialog box opens.

2. Mark the desired attribute set in the left window pane.

3. Choose *Add attribute* on the context menu or click the corresponding icon.

   Icon for adding an attribute

   The *Add attribute* dialog box opens.

4. Mark the attributes that you want to add and click the *OK* button.

   The selected attributes are added to the attribute set.

5. Click the *Apply* button.

Removing Attributes from an Attribute Set

To remove an attribute from an attribute set:

1. Choose *Tools* → *Website administration* → *Attribute sets*.

   The *Attribute sets* dialog box opens.

2. Mark the desired attribute set in the left window pane.

3. In the right window pane, mark the attributes that you want to remove from the attribute set.

4. Choose *Remove attribute* on the context menu or click the corresponding icon.

   Icon for removing an attribute from an attribute set

5. Click the *Apply* button.

   The attributes are removed from the set, but remain in the database. Thus, they can later be assigned again.

Deleting an Attribute Set

> **Note:** You can only delete attribute sets that are not used by any object in the relevant website.

To delete an attribute set:

1. Choose *Tools* → *Website administration* → *Attribute sets*.

   The *Attribute sets* dialog box opens.

2. Mark the desired attribute set in the left window pane.
3. Choose *Delete attribute set* on the context menu or click the corresponding icon.

   Icon for deleting an attribute set

4. Confirm the security prompt by clicking the *Yes* button.

Managing Object Categories

For general information on object categories, refer to section “Properties from the object category” on page 410 and to the Livelink WCM Server Administrator Manual.

When adding a WCM object, you can optionally assign an object category to the object. It then possesses the properties that were defined for this category. These properties supplement the standard metadata of WCM objects and are displayed on the *Properties* tab in the Metadata dialog box.

By means of WCM tags, the values of these properties can be automatically entered in the generated pages and, for example, queried when searching a website. Object categories also form the basis for assembled objects (Content Assembly).

To view the object categories of the current website, choose *Tools* → *Website administration* → *Object categories*. The *Object categories* dialog box opens. The object categories are displayed in a list in the left window pane. To view the properties of an object category, select the object category in the list. Immediately after the installation of Livelink WCM Server, there are no object categories.

The following functions are available to you:

- adding object categories, see the following section
- adding properties to existing object categories or deleting properties, see section “Editing an Object Category” on page 487
- deleting object categories, see section “Deleting an Object Category” on page 490

Note: If the proxy Content servers assigned to the respective website use separate data storages, changes to object categories must also be saved there. For this reason, the system checks whether these proxy Content servers are available and whether the JDBC pools assigned can be used. If this is not the case, you cannot save the changes to the object categories. Please refer to the Livelink WCM Server Administrator Manual for additional information.

Adding an Object Category

To add an object category:

1. Choose *Tools* → *Website administration* → *Object categories*.

   The *Object categories* dialog box opens.

2. Choose *New object category* on the context menu or click the corresponding icon.

   Icon for adding an object category

   The *New object category* dialog box opens.

3. Enter a name for the new object category. The name of the object category must be unique within the website and must not be more than 17 characters long. The following characters are permitted:

   - a-z and A-Z (no umlauts and special characters)
   - 0–9, the first character must always be a letter
   - _ (underline)

   Note: The following names are reserved and may not be used: admin, acl, ats, att, cda, con, dyn, lat, lin, obj, oty, par, pri, pro, pty, rli, sta, tta, tto.

You cannot change the name subsequently.

4. Add the object category by clicking the *OK* button.
5. Define the properties for the new object category (see the following section).

Editing an Object Category

Note: When object categories are changed, the pages of the WCM objects to which the category is assigned are **not** automatically regenerated. In this case, it may happen that the values of deleted properties are still displayed on the generated pages.

The following functions are available for editing object categories:

- creating properties, see the following section
- deleting properties, see section “Deleting Properties” on page 489

Creating Properties

Notes:

You cannot change existing properties subsequently.

You cannot reuse the properties of existing object categories; you must create new ones. If you want to use the same property, e.g. “Price”, for two different object categories, you must create this property twice with different names, but identical settings.

To create a property for an object category:

1. Choose *Tools* → *Website administration* → *Object categories*.

   The *Object categories* dialog box opens.
2. Mark the desired object category in the left window pane.

3. Choose *New property* on the context menu or click the corresponding icon.

   Icon for creating a property

   The *New property* dialog box opens.

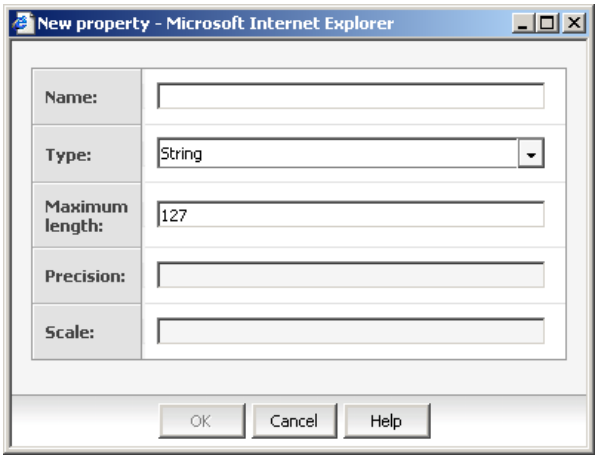


Fig. 136 – Dialog box when creating a property

4. Make the following settings:

   - *Name:* The name must be unique within the entire website. Moreover, the name of the property must not be identical to the name of an attribute from an attribute set in this website.

     The name must not be more than 20 characters long. Otherwise, the same conditions as for names of categories apply.

     Note: If you want the name of the property to be displayed in the language of the user, you can translate it. For more information, refer to the Livelink WCM Server Administrator Manual.

- *Type:* Click a type in the drop-down list. For detailed information on the individual types, refer to table “Data types with formats used” on page 436.
- *Maximum length:* For the types “External URL”, “List”, “Set”, “Set of e-mail addresses”, and “String”, you must determine the maximum length. Select a high value because, for example, in the Oracle RDBMS some special characters are represented by 2 bytes instead of 1 byte. The maximum length of character strings is determined by the RDBMS used (Oracle: 4000 characters).
- *Precision* (only for the type “Floating-point number”)*:* determines the number of digits to be considered
- *Scale* (only for the type “Floating-point number”): determines the number of digits after the decimal point, i.e. where the point is inserted in the numeric string

  The value range for the type “Floating-point number” is defined as follows: from $-(10^{p-s}-10^{-s})$ to $+(10^{p-s}-10^{-s})$,
  with p = precision and s = scale
  Example: A precision of 5 with a scale of 3 results in a value range of –99.999 to +99.999.

5. Click the *OK* button.

   The new property is added to the object category.

6. Click the *Apply* button.

Deleting Properties

To delete properties of an object category:

1. Choose *Tools* → *Website administration* → *Object categories*.

   The *Object categories* dialog box opens.

2. Mark the desired object category in the left window pane.

3. Mark the properties to be deleted in the right window pane.
4. Choose *Delete property* on the context menu or click the corresponding icon.

   Icon for deleting a property

5. Click the *Apply* button.

Deleting an Object Category

> **Note:** You can only delete object categories that are not used by any object in the relevant website.

To delete an object category:

1. Choose *Tools* → *Website administration* → *Object categories*.

   The *Object categories* dialog box opens.
2. Mark the desired object category in the left window pane.
3. Choose *Delete object category* on the context menu or click the corresponding icon.

   Icon for deleting an object category

4. Confirm the security prompt by clicking the *Yes* button.

APPENDIX A

Livelink WCM Server and XML

XML is especially suitable for exchanging platform-independent data between different systems or for separating content and layout. For this reason, XML is often used in content management environments.

Livelink WCM Server provides four object types for integrating and editing XML data structures in a WCM-managed website. This appendix describes how to use these XML object types. For this chapter, a sound knowledge of XML and Livelink WCM Server is required.

> **Note:** For the XML object types to work correctly, you may have to create a MIME type for XSL documents in the application server. If this is necessary, contact your WCM system administrator.

The XML Object Types

Livelink WCM Server provides four object types that you can use to add XML objects to your website:

- "XML document"
- "XML template"
- "XSLT document"
- "XSLT template"

These object types are introduced in the following sections.

Note: In theory, you can add the object type “XML topic” via the Admin client or the Content client. However, Livelink WCM Server does not support such an object type at present.

The Object Type “XML document”

The object type “XML document” is used to add XML data structures as WCM objects. For the Edit and QA views of XML documents, the WCM system generates a so-called *surrogate page*, which contains a link to the XML document. In the Production view, however, the content of the XML document is visible.

Sample code for an XML document

```
<?xml version="1.0" encoding="UTF-8"?>

<product>
      <name>software package</name>
      <order_no>123456789</order_no>
      <price>1250 euro</price>
      <description>
         The all-in-one package for developers. Out
         now.
      </description>
</product>
```

The Object Type “XML template”

An object of type “XML template” is used as a template and has all the properties of an XML document. Thus, you can assign an XML document an XML template in Livelink WCM Server. This is, for example, useful if you manage many XML documents with a similar structure. Code segments contained in all XML documents can be offloaded into an XML template.

During page generation, the content of the XML document is mixed with the content of the XML template. The following must be considered for XML: if the `<?xml` tag is located in the first line of the XML document, it is not integrated in the content of the generated page. In this case, Livelink WCM Server assumes that such a tag already exists in the XML template.

Sample code for an XML template

```
<?xml version="1.0" encoding="UTF-8"?>

<product>
      <name>product template</name>
      <order_no>{VIPORDER_NO}</order_no>
      <price>{VIPPRICE}</price>
      {VIPCONTENT}
</product>
```

The Object Type “XSLT document”

An object of type “XSLT document” is used to define transformation rules that are applied to XML documents. This way you can easily format the content of an XML document.

Sample code for an XSLT document

```
<?xml version="1.0" encoding="UTF-8"?>

<xsl:stylesheet version="1.0"
      xmlns:xsl="http://www.w3.org/1999/XSL/Transform">

<xsl:output method="html" version="1.0" encoding="UTF-8" indent="yes"/>

<xsl:template match="product">
<html>
<head>
      <title><xsl:value-of select="name"/></title>
</head>
<body>
```

```
    <table width="40%">
      <tr>
          <th align="left" colspan="2"><xsl:value-of select=
            "name"/></th>
      </tr>
      <tr>
          <td>order no:</td>
          <td><xsl:value-of select="order_no"/></td>
      </tr>
      <tr>
          <td>price:</td>
          <td><xsl:value-of select="price"/></td>
      </tr>
      <tr>
          <td colspan="2"><em><xsl:value-of select=
            "description"/></em></td>
      </tr>
    </table>
</body>
</html>
</xsl:template>

</xsl:stylesheet>
```

You reference the XSLT document as a style sheet directly in an XML document or XML template, for example in the form:

```
<?xml-stylesheet type="text/xsl" href="transformation.xsl"?>
```

The transformation rules contained in the XSLT document are applied to the XML document (or to all XML documents to which the XML template is assigned). The transformation of the XML data by the style sheet is performed on the client side, i.e in the browser.

Note: XSLT is not supported by all browsers. Thus, the possible applications of XSLT documents are limited.

The Object Type “XSLT template”

An object of type “XSLT template” is used as a template and has all the properties of an XSLT document.

XSLT templates produce usable results if the following requirements are met:

- Only assign XSLT templates to objects of type “XML document”.
- Do not use XSLT templates in cascading XML templates.

The transformation rules defined in the XSLT template are applied to those XML documents that have been assigned the template. In contrast to the XSLT document, transformation is performed on the server side, using special cache mechanisms.

> **Note:** If they do not contain WCM tags, XSLT templates are saved in the cache. If the template references a style sheet by means of `xsl:include` or `xsl:import` and this style sheet changes, the template is not automatically removed from the cache. In this case, regenerate the XSLT template.

The code of an XSLT template does not differ from the code of an XSLT document that is based on the same file. Thus, sample code is not included here.

Assign an HTML template, JSP template, or XML template to the XSLT template. This gives the transformation a framework. In this way, the result of the transformation is integrated in a defined page structure and has the desired file extension. If you do not assign a template to the XSLT template, the result of the transformation has the file extension **.xsl**. When calling an XSL file, the browser expects an XML style sheet. However, the result of the transformation contains HTML code. Thus, the browser cannot display the content.

XSLT templates can be used in many ways. Like an XSLT document, you can use an XSLT template to format the content of an XML document. In the XSLT template, it is also possible to define conditions that dynamically control the layout of the generated data. You can define these conditions on the basis of WCM tags, e.g.

```
<xsl:when test="'{VIPDEPLOYMENT_NAME}'='{VIPSITE}_prod'">
```

Using Templates

Like any other object, objects of type “XML document” can be assigned templates. Depending on the type of the template assigned, the content and file extension of the generated page change. The following table gives you an overview. The table includes only those combinations of templates that have usable results.

Table 49 – Content and file extension depending on the template

Template	Generated page	
	Content	**File extension**
none	The complete content of the XML document is output including the `<?xml` tag.	**.xml**
HTML or JSP template	The content of the XML document – without the `<?xml` tag – is inserted at the location marked by `{VIPCONTENT}`.	**.html** or **.jsp**

Template	Generated page	
	Content	**File extension**
XML template	The `<?xml` tag is adopted from the template. The content of the XML document – without the `<?xml` tag – is inserted at the location marked by `{VIPCONTENT}`.	**.xml**
XML template that has been assigned an HTML template or a JSP template	The `<?xml` tag is adopted from the template. The content of the XML document – without the `<?xml` tag – is inserted in the XML template at the location marked by `{VIPCONTENT}`. Then this content is inserted in the HTML or JSP template at the location marked by `{VIPCONTENT}`.	**.html** or **.jsp**
XSLT template that has been assigned an HTML template or a JSP template or an XML template	For all WCM tags recognized, the respective values are inserted in the XML document and in the XSLT template. After the XSLT transformation, the result of the transformation is inserted in the HTML, JSP, or XML template at the location marked by `{VIPCONTENT}`.	**.html**, **.jsp**, or **.xml**

Application Examples

On the basis of two application examples, the interaction of the XML object types is described in this section.

Transformation of XML Data on the Client Side

This example explains how to display product information stored in an XML file in a clearly structured way on an HTML page by means of an XSL file. The transformation of the XML data to HTML is performed on the client side, i.e. in your browser.

Step 1 – Create the XML document

Your product information is saved in the file **product.xml**. Use the Content client to add an XML document based on this file.

Code of the file **product.xml**

```
<?xml version="1.0" encoding="UTF-8"?>

<product>
    <name>software package</name>
    <order_no>123456789</order_no>
    <price>1250 euro</price>
    <description>
         The all-in-one package for developers. Out
         now.
    </description>
</product>
```

Step 2 – Create XSLT document

Your transformation rules are defined in the file **transformation.xsl**. Use the Content client to add an XSLT document based on this file.

Code of the file **transformation.xsl** for transformation on the client side

```
<?xml version="1.0" encoding="UTF-8"?>

<xsl:stylesheet version="1.0"
      xmlns:xsl="http://www.w3.org/1999/XSL/Transform">

<xsl:output method="html" indent="yes"/>

<xsl:template match="product">
<html>
<head>
      <title><xsl:value-of select="name"/></title>
</head>
<body>
    <table width="40%">
      <tr>
          <th align="left" colspan="2"><xsl:value-of select=
            "name"/></th>
      </tr>
      <tr>
          <td>order no:</td>
          <td><xsl:value-of select="order_no"/></td>
      </tr>
      <tr>
          <td>price:</td>
          <td><xsl:value-of select="price"/></td>
      </tr>
      <tr>
          <td colspan="2"><em><xsl:value-of select=
            "description"/></em></td>
      </tr>
    </table>
</body>
</html>
</xsl:template>

</xsl:stylesheet>
```

Step 3 – Reference the XSLT document in the XML document

The transformation rules contained in the XSLT document are to be applied to the XML document. In the XML document, add the following reference to your XSLT document:

```
<?xml-stylesheet type="text/xsl" href="transformation.xsl"?>
```

In the Production view, the HTML page generated from the XML document might look as follows:

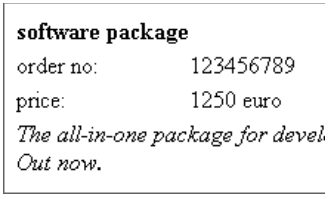


Fig. 137 – Result of the transformation on the client side

As transformation is performed on the client side, the source code has the format XML.

Transformation of XML Data on the Server Side

This example explains how to display product information stored in an XML file in a clearly structured way on an HTML page by means of an XSL file. In this example, the transformation of the XML data into HTML is performed on the server side.

The two files **product.xml** and **transformation.xsl** from the first application example are the starting point again. For a transformation on the server side, the head element and both the body tags and HTML tags must be removed from the XSLT template. Modify the file **transformation.xsl** accordingly.

Code of the file **transformation.xsl** for transformation on the server side

```
<?xml version="1.0" encoding="UTF-8"?>

<xsl:stylesheet version="1.0"
      xmlns:xsl="http://www.w3.org/1999/XSL/Transform">

<xsl:output method="html" indent="yes"/>

<xsl:template match="product">
      <table width="40%">
      <tr>
          <th align="left" colspan="2"><xsl:value-of select=
            "name"/></th>
      </tr>
      <tr>
          <td>order no:</td>
          <td><xsl:value-of select="order_no"/></td>
      </tr>
      <tr>
          <td>price:</td>
          <td><xsl:value-of select="price"/></td>
      </tr>
      <tr>
          <td colspan="2"><em><xsl:value-of select=
            "description"/></em></td>
      </tr>
    </table>
</xsl:template>

</xsl:stylesheet>
```

Perform the following steps:

1. Create an XML document based on the file **product.xml**.
2. Create an XSLT template based on the file **transformation.xsl**.
3. Create an HTML template without specifying a file.

   The code of this template object looks as follows:

```
<html>
<head></head>
<body>{VIPCONTENT}</body>
</html>
```

4. Assign the HTML template to the XSLT template.

   With this assignment, you determine the format of the file which is generated when the XSLT template is assigned to an object (in this case HTML).

5. Assign the XSLT template to the XML document.

   The XML document is transformed on the server side. For this purpose, an XSLT processor is integrated in the WCM server.

   The code of the HTML page generated from the XML object looks as follows:

```
<html>
<head>
</head>
<body>
<table width="40%">
<tr>
<th colspan="2" align="left">software package</th>
</tr>
<tr>
<td>order no:</td><td>123456789</td>
</tr>
<tr>
<td>price:</td><td>1250 euro</td>
</tr>
<tr>
<td colspan="2"><em>The all-in-one package for developers. Out
   now.</em></td>
</tr>
</table>
</body>
</html>
```

In the Production view, the HTML page looks as follows:

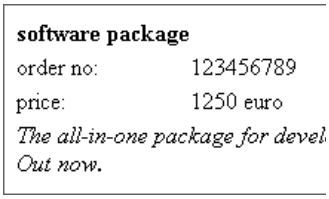


Fig. 138 – Result of the transformation on the server side

As transformation is performed on the server side, the source code has the format HTML.

Glossary

Access control list – For each WCM object, users, groups, roles, and group-roles that have access to this object can be specified. The individual access rights are specified separately for each principal authorized to access the object. Also called ACL.

ACL – Access Control List

Activity – Workflow step to which a principal (user, group, or role) is assigned. An activity step represents the tasks of the assigned principal. There are three types of activities: “Edit”, “QA”, and “Delete”.

Address – see *URL*

Applet – Java program embedded in a website. An applet is loaded by the server and executed by the client.

ASP – Active Server Pages. HTML files with specifically identified embedded JavaScript or Visual Basic Script programs that are run on the web server. The result is then sent to the client in normal HTML format.

Asynchronous action – An asynchronous action returns almost immediately after it has been called. It does not wait for processing of the associated tasks to finish. See also *Synchronous action*.

Attributes – Special metadata that can be defined differently for each object type. Attributes are grouped in attribute sets.

Attribute set – Set of attributes. Attribute sets can be combined with object types. In this way, all WCM objects that are created on the basis of this object type can be equipped with the attributes of the assigned attribute set. Attribute sets are edited in the Admin client or in the Content client.

Browser – A browser is primarily an application program for displaying HTML pages. With the increasing spread of Internet technologies, the browser is becoming a universal client and serves as an interface for all

information and applications based on Internet technologies. In the context of Livelink WCM Server, the browser serves as an interface to the Edit, QA, and Production view.

CGI – Common Gateway Interface. A web server interface used to run scripts or programs that generate user responses on HTML forms. CGI programs are usually located in a special directory on the HTTP server. Special URLs call such a CGI program, which in turn generates an HTML response to the request and sends it to the client.

Check in – Staging action of Livelink WCM Server. An object that has been checked out and edited is returned to the WCM system by the action “Check in”. This makes the changes visible in the Edit view. The object is combined with the template again and is available for further editing.

Check out – Staging action of Livelink WCM Server. The content of a WCM object can only be edited after the object has been checked out. A checked-out object is locked for access by other users. The changes to the object do not become available in the Edit view until the object has been checked in.

Client – Process, program, or computer for requesting a service from a server. The browser, for example, is a client program used to request HTML pages from a web server.

Cookie – Information sent by a web server to the browser. The browser saves the information in a text file. In this way, statuses can be saved and the user will see his/her familiar environment when accessing the Internet page again. When the user visits the page again, the browser sends a copy of the cookie to the web server. For security reasons, the information of a cookie is returned only to the web server that originally sent the cookie.

Deployment – Deployment is the distribution of data. The deployment of Livelink WCM Server performs two main tasks: first, generating pages from the WCM objects stored in the database and distributing the

generated files to the appropriate directories; second, notifying the WCM servers of changes in the WCM system.

Deployment system – The deployment systems generate pages from the WCM objects and distribute the generated files to the appropriate directories. From there, the files become visible for the users via an HTTP server. Deployment systems may be of various types and categories.

Edit view – In the Edit view of Livelink WCM Server, the objects of a website are created and edited. Here the most current status of the objects is visible.

End point – When a WCM object is forwarded in the workflow and an end point is reached, the assignment between WCM object and workflow is removed.

Extranet – Information platform based on Internet technology for business communication with authorized external users, e.g. partners or customers.

Form – Possibility to integrate dialog elements to be filled in by users in web pages. The content of such a form can be sent to the server for further processing.

Frame object – Object type for a frameset

Frame topic – Frame object that is at the same time used as a topic (for content organization). See also *Topic*.

Group – Collection of users for which specific access rights can be defined. User groups are usually tied to organizational units, such as departments and projects.

Group-role – Combination of a group and role. The group-role is not an organizational unit that is defined in the Admin client, but a configuration option for the object access rights in the Content client.

Home page – The start page of a website. It usually performs the function of a title page and table of contents for the website.

HTML – Hypertext Markup Language. A text description language for creating pages that can be displayed in the browser.

HTML page – HTML file that is displayed by the browser

HTTP – Hypertext Transfer Protocol. A communication protocol for transferring HTML pages

Java – Object-oriented programming language developed by Sun Microsystems and used especially in the field of Internet technology. Security aspects and platform independence are the basic philosophies of Java.

JavaScript – Scripting language for integrating executable scripts in HTML pages. JavaScript can be used to incorporate plausibility checks and calculation functions in forms.

Metadata – Every WCM object has a number of object information assigned to it (e.g. expiration date, language). These are known as metadata.

Object – Each element of a website is integrated in Livelink WCM Server as a single object. Each object is based on an object type, such a “HTML page”. Object types can be defined in the Admin client or the Content client.

Object category – Assignment of a WCM object to a specific category. Due to this assignment, the WCM object has a set of additional special attributes (metadata).

Object fragment – Metadata item integrated into the generated page by means of a WCM tag. In InSite Editing, fragments are highlighted by a red frame and can be edited separately without checking out the WCM object.

Object type – The specific kind of object, e.g. “HTML page”, “HTML template”, “Topic”. Various properties of the WCM object result from the object type. The object type is defined when the object is created. There are only a few cases in which it may subsequently be changed. Object types can be edited in the Admin client or the Content client.

Portal – A portal is a website that serves the user as a central point of access – as a gate – to certain Internet services. A portal often offers topic-specific and personalized offers and information.

Production view – The Production view of Livelink WCM Server makes the released pages of a website available to the user. By means of a web server, these pages can be accessed in the Internet, intranet, or extranet.

QA view – The QA view of Livelink WCM Server is used for quality assurance of the objects and thus of the website content. This view thus performs the control function between editing in the Edit view and publication in the Production view.

Reference – Cross-references and links managed by Livelink WCM Server between the objects in a website are known as references. Only references that are physically incorporated in the HTML page are called hyperlinks.

Relator – Due to the integration of web content management and Livelink, you can add Livelink objects as WCM objects in a WCM-managed website. These WCM objects are called relators. They always refer to a certain version of a Livelink object. For integrating a single Livelink object, the WCM system provides the object type “Livelink relator”. For integrating Livelink folders, the WCM object type “Livelink folder relator” is provided.

Release – Staging action of Livelink WCM Server. Quality Assurance checks whether the content and the form of a submitted object meet the quality standards of the company. If so, the object is released. The release transmits the quality-assured version of the object to the Production view, and thus makes it available to the end user in the published website.

Role – Collection of users, similar to a user group, for which specific access rights can be defined. The user role is usually defined in terms of tasks, whereas user groups are generally tied to organizational units, such as departments or projects.

Server – Process, program, or computer for handling the requests from a client and for providing services that can be used by a client.

Starting point – Initial point of a workflow. When a workflow is assigned to a WCM object, the object is automatically forwarded to the first activity.

Statification – During statification, the dynamic components of, for example, a JSP page are converted into static components. The result is pure HTML without Java code.

Submit – Staging action of Livelink WCM Server. before a newly created or changed object can be published, it must be submitted to Quality Assurance for checking. This makes the changes to the object visible in the QA view.

Synchronous action – A synchronous action does not return until all the tasks involved are completed. See also *Asynchronous action*.

Template – File that provides the framework for other objects. The HTML pages for the Production view are created by bringing together the template and the content. Any HTML components may be integrated in a template.

Topic – Combination of an HTML page for direct display of information and a list for accommodating subordinate objects. The topics are shown as nodes in the object tree and thus help structure the website.

Transition – Connection between two activities or a starting/end point and an activity in a workflow. By means of the transitions, the WCM objects are forwarded within the workflow.

Type – see *Object type*

URL – Uniform Resource Locator. A unique address in the World Wide Web.

WCM server – In a WCM system, there are several WCM servers working in parallel (server processes). The exact tasks of a server depend on the server type and server category.

WCM tag – Special element for integrating WCM specific data in an HTML page. WCM tags are used in particular when creating templates.

Web browser – see *browser*.

WebDAV – The WebDAV (Web-based Distributed Authoring and Versioning) protocol supports Internet and group-based working on the basis of standard Internet technologies. Thanks to WebDAV, users do not need a special client for creating WCM objects, jointly editing them, and managing them by means of WevDAV-capable tools.

Web repository – The name given to the area where the WCM system stores and manages all the objects belonging to a website

Web server – Program for processing the requests from a browser. A web server corresponds to an HTTP server that offers additional web services, such as those of a JSP engine.

Web technology – Client/server technology based on TCP/IP. The advantage lies in the open standard, which permits the creation of heterogeneous systems.

Workflow definition – Description of a workflow. A workflow definition consists of starting points, end points, activities, and transitions. Workflow definitions are saved in XPDL format and can be edited graphically by means of the Content Workflow Modeler.

WWW – World Wide Web. In the WWW, HTML documents are linked with each other by means of URLs. It is one of the most frequently used Internet services.

WYSIWYG – What you see is what you get. A buzzword for the property of files that enables you to see on the display screen (almost) exactly what will appear when the document is printed.

XML – Extensible Markup Language. A standard for defining the formal structure of documents.

XSL – Extensible Stylesheet Language. A standard for commands used for converting XML documents to other formats. XSL is often used for converting XML to HTML.

Index

A

absolute references 48
Absolute URL (special attribute) 416
access control list
  add principal 216
  remove principal 222
access rights
  add principal 216
  and actions 50
  default settings for 54
  edit 215
  for the Production view 225
  in general 50
  inheritance 55
  interpret 54
  priority rules 57
  remove principal 222
  reset all 226
  set 220
  set for all users 223
Access rights dialog box 215
action list
  available functions 102
  configure 119
  in general 101
ActiveX 112
activity
  change size 268
  delete 270
  move 267
add
  application class 122
  attribute 480
  attribute set 479
  attribute to attribute set 483
  cascading template 405
  folder relator 346
  Livelink object as WCM object 339
  object 134
  object based on object type 134
  object based on template 148
  object category 486
  object type 467
  profile 109
  reference in Content client 204
  reference in editor 209
  reference in integrated HTML editor 206
  relator 341
  relator with Livelink document 350
  template 399
application class 121
  add 122
  delete 124
  edit 124
ASP page (object type) 66
ASP template (object type) 66
ASP topic (object type) 66
assembled object
  application examples 324
  determine default values 328
  edit 324
  in general 317
  object data type 471
  object type 66
assembled object (create)
  add object category 318
  create template 322
assign
  workflow 282
asynchronous actions 101
attribute
  add to attribute set 483

create 480
remove 483
attribute set 409
add 479
add attribute 483
assign to object type 475
delete 484
edit 480
functions 478
manage 477
new attribute 480
remove attribute 483
author
metadata item 43
WCM tag 429

B

basic settings
configure 110
boolean value (data type) 436
broken links 203

C

cascading template 401
add 405
delete 406
edit 405
release 405
structure 402
CGI script (object type) 66
change password 78
changed (object status) 35
check in 174
check out
object 168
undo 173
checked out (object status) 35
client 27
client-server technology 27
combination operators 390
comparators 385
comparison values 388
compound object
convert 239
delete 241
edit 240
metadata 242
object data type 471
object type 66
special properties 242
submit 241
configuration 107
action list 119
application class 121
basic settings 110
Download applet 113
dynamic deployment 125
editor program for file types 121
InSite Editing 309
integrated HTML editor 114
navigation 117
object attributes 120
toolbar 116
Configuration dialog box 108
configuration file for InSite Editing 309
content
display 89
edit 178
edit via InSite Editing 297
in general 39
Content Assembly
application examples 324
determine default values 328
enable InSite Editing 332
in general 317
with Content client 325
with InSite Editing 331
Content client
configuring 107
customizing 129

help 105
in general 73
introduction 18
log in 75
log out 77
user interface 81
Content Workflow Modeler
functions 265
in general 262
install 261
menu bar 265
start 263
toolbar 265
convert
compound object 239
in general 233
object 239
copy
dependent objects 236
object 162
source object 236
topic 164
country code (WCM tag) 429
create
filter 377
property 487
transition 268
workflow definition 272
created on
metadata item 43
WCM tag 429
CSS (object type) 66
current object 86
customizing 129
through access rights 129
through functional areas 130
through profiles 130
cut 187

D

data type
boolean value 436
date 436
external URL 436
floating-point number 436
formatted text 436
integer 437
large integer 437
list 437
Livelink ID 437
object content 438
object source 438
object URL 438
OID 438
set 437
set of e-mail addresses 438
string 438
text 439
time 439
time stamp 439
date (data type) 436
delayed release
add object 139
metadata item 43
object status 35
set for an object 198
WCM tag 429
delete
activity 270
application class 124
attribute set 484
cascading template 406
compound object 241
end point 270
entry from action list (automatically) 120
entry from action list (manually) 102
filter 385
object category 490
object not yet released 229

object type 477
page 232
profile 128
property 489
reference 214
reject 231
relator 362
released object 227
source object 241
starting point 270
template 400
transition 270
workflow definition 280
workflow element 270
deleted (object status) 36
dependent object 248
deployment
dynamic 90
description
metadata item 43
WCM tag 429
destroy (object) 230
dialog
metadata dialog box 193
dialog box
Access rights dialog box 215
Change password 78
Configuration dialog box 108
Log in 75
Login information 80
References dialog box 202
Select website 76
Send e-mail 92
Take over as substitute 79
direct release 253
add object 139
metadata item 43
WCM tag 429
display
content 89
information on workflow 92
preview in browser 89
preview in editor 92
Download applet 113
configure 113
dynamic deployment
configure 125
object preview 90
dynamic deployment system 90

E

edit
application class 124
assembled object 324
attribute set 480
cascading template 405
checked-out object 184
compound object 240
content of a relator 354
filter 383
Livelink metadata 355
metadata of several objects 196
object category 487
object content 178
object type 475
profile 128
several objects 365
source object 240
special attributes 411
template 400
workflow definition 278
Edit view 29
editor 37
configure for file types 121
editor program
for file types 121
e-mail
send 92
with link to Content client 94
E-mail Edit
add object 141

metadata item 43
WCM tag 429
E-mail QA
add object 141
metadata item 43
WCM tag 429
E-mail Release
add object 141
metadata item 43
WCM tag 429
embedded object 146
end point
change size 268
delete 270
move 267
examples for filter expressions 393
Excel document (object type) 66
expiration date
add object 139
WCM tag 429
external references 47, 203
external URL (data type) 436

F

favorites list 85
file extension (WCM tag) 429
file types
configure editor program 121
filter
add 377
delete 385
edit 383
use variables 382
filter editor 376
add filter 377
filter expression language 385
combination operators 390
comparators 385
comparison values 388
examples 393
nested filters 392
search for principals 391
search for workflow assignment 392
filter functions
introduction 96
use 365
filter variable 382
Fixed version
folder relator 349
relator 345
floating-point number (data type) 436
folder relator 339
add 346
form instance (object type) 67
form template (object type) 67
formatted text (data type) 436
fragment 299
edit via InSite Editing 298
frame
object data type 471
object type 67
frame topic
object data type 471
object type 67
functional area 61
access rights dialog 63
Advanced 62
assign to object type 475
Basic 62
Direct release 63
Dynamic 62
Filter edit 64
Filter standard 63
Form 62
import 64
Intelligent templates 63
Livelink 63
log dialog 63
References dialog 63

Search 64
Template structure 64
View My objects 64
View Object list 64
View Subordinate objects 64
Workflow 62
functions
attribute sets 478
InSite Editing 296
object types 467
workflow modeler 265

G

generate_static (special attribute) 415
GIF image (object type) 67
group 51
add to access control list 216
set access rights 220
withdraw access rights 222
group-role 52

H

header 81
heading
metadata item 44
WCM tag 429
help 105
hierarchy level (WCM tag) 429
HTML page (object type) 67
HTML template
object data type 471
object type 67
hyperlink
add in Content client 204
add in editor 209
add in integrated HTML editor 206
delete 214
in general 46
maintain 201

I

import from ZIP file 158
select file 159
specify title and object type 161
start wizard 159
import linked objects 154
select file 156
start wizard 155
InSite Editing
application example 304
call functions 296
configuration file 309
configure 309
edit object content 297
edit object fragments 298
enable for metadata item 183
for assembled objects 332
functions 296
in general 293
start 295
switch off for individual pages 314
WCM tag dialog box 299
install
workflow modeler 261
integer (data type) 437
integrated HTML editor 114
add reference 206
configure 114
insert WCM tag 179
invalid references 203

J

Java Web Start
activate 112
Workflow modeler 263
JavaScript (object type) 67
JPEG image (object type) 67
JSP page (object type) 68
JSP template (object type) 68
JSP topic (object type) 68

K

keywords
metadata item 44
WCM tag 429

L

language
metadata item 44
WCM tag 429
language code (WCM tag) 429
large integer (data type) 437
later release
set for an object 198
licenses 77
link
add in Content client 204
add in editor 209
add in integrated HTML editor 206
delete 214
in general 46
maintain 201
list (data type) 437
Livelink
functional area 63
Livelink document
add with relator 350
Livelink folder relator 339
add 346
object type 68
Livelink ID 344
data type 437
Livelink metadata
edit 355
integrate in page 358
WCM tags 358
WCM tags for standard metadata 358
WCM tags for user-defined metadata 359
Livelink metadata dialog box 355
Livelink object
add as WCM object 339
Livelink relator 339
add 341
add with document 350
delete 362
edit content 354
edit metadata 355
integrate metadata 358
object data type 471
object type 68
update manually 361
Livelink WCM Server
staging 29
system architecture 27
workflow 30
log 103
log in 75
log out 77
login information 80

M

major version 87
manage
attribute sets 477
object categories 485
object types 466
websites 465
menu
favorites 85
menu bar
Content client 81
workflow modeler 265
metadata
compound object 242
edit 193
edit metadata of several objects 196
in general 42
integrate via WCM tags 428

source object 236
special attributes 409
standard metadata 43
micro version 87
MIME type 472
minor version 87
modification date
metadata item 44
WCM tag 429
modified by
metadata item 44
WCM tag 429
move
activity 267
dependent objects 236
end point 267
object 187
source object 236
starting point 267
transition 269
my objects
introduction 97
use 366
my worklist
view 98

N

navigation
configure 117
navigation area 81
new
application class 122
attribute 480
attribute set 479
cascading template 405
filter 377
folder relator 346
Livelink object as WCM object 339
object 134
object based on object type 134
object based on template 148
object category 486
object type 467
profile 109
property 487
reference in Content client 204
reference in editor 209
reference in integrated HTML editor 206
relator 341
relator with Livelink document 350
template 399
new object
define object responsibilities 140
enter remark 146
select file 135
select workflow 135
select workflow transition 145
specify basic properties 137
specify title and object type 135
specify values for attributes 142
specify values for properties 144
start wizard 135
notification
send 92

O

object
activate thumbnails 422
add 134
assign workflow 282
call functions in InSite Editing 296
change type 199
check in 174
check out 168
convert 239
copy 162
cut 187
delete (not yet released) 229
delete (released) 227
dependent 248

destroy 230
edit again 184
edit content 178
edit in workflow 285
edit metadata 193
embedded 146
find 365
insert WCM tag 179
integrate metadata 200
log 103
move 187
reject 255
release 251
release directly 253
replace 190
restore older version 186
set delayed release option 198
submit 248
undo changes 186
undo check out 173
undo submit 250
update 189

object category
add 486
add object 139
delete 490
delete properties 489
edit 487
in general 410
manage 485
metadata item 44
new property 487
WCM tag 429

object content 89
data type 438
edit via InSite Editing 297
integrate via WCM tags 428
replace 190

object data type 470
Assembled object 471
Compound object 471
Frame 471
Frame topic 471
HTML template 471
Livelink relator 471
surrogate 471
Topic 471
web document 471
workflow 472
XML document 472
XML template 472
XSLT template 472

object filter
introduction 96
use 366

object fragment 299
edit via InSite Editing 298
simple 299
with object reference 299

object list 84

object preview
dynamic deployment 90
in general 89

object reference 299

object rights
in general 50

object size (in bytes)
metadata item 44
WCM tag 430

object source (data type) 438

object status 32
changed 35
checked out 35
delayed release 35
deleted 36
rejected 36
released 36
submitted 36

object type
add 467
ASP page 66
ASP template 66

ASP topic 66
Assembled object 66
CGI script 66
change 199
Compound object 66
CSS 66
delete 477
edit 475
Excel document 66
Form instance 67
Form template 67
Frame 67
Frame topic 67
functions 467
GIF image 67
HTML page 67
HTML template 67
in general 65
JavaScript 67
JPEG image 67
JSP page 68
JSP template 68
JSP topic 68
Livelink folder relator 68
Livelink relator 68
manage 466
Other 68
PDF document 68
PHP page 68
PHP template 68
PHP topic 69
PNG image 69
PowerPoint file 69
Script 69
Topic 69
Word document 69
Workflow 69
XML document 69
XML schema 70
XML template 70
XSLT document 70
XSLT template 70
object URL (data type) 438
OID
  data type 438
  metadata item 45
  WCM tag 430
old version
  restore 186
online help 105
operators for searching principals 391
Other (object type) 68

P

page
  merge template and object 398
  remove from Production view 232
parameter for WCM tags
  add 461
  buttons 460
  default 452
  edit 458
  encoding 450
  filter 451
  format 441
  label 453
  locale 449
  oid 457
  option_id 454
  option_mode 455
  order 459
  required 453
  separator 448
  target 461
  thumbnails 452
  type 450
parameterized WCM tags 440
parent topic (WCM tag) 430
path (metadata item) 45
PDF document (object type) 68
PHP page (object type) 68
PHP template (object type) 68

PHP topic (object type) 69
PNG image (object type) 69
PowerPoint file (object type) 69
preview
  in browser 89
  in editor 92
principal 37
  add to access control list 216
  group 51
  group-role 52
  role 51
  set access rights 220
  withdraw access rights 222
  World 52
priority rules (access rights) 57
Production view 29
profile 107
  action list 119
  add 109
  application class 121
  basic settings 110
  delete 128
  Download applet 113
  dynamic deployment 125
  edit 128
  name 109
  navigation 117
  object attributes 120
  toolbar 116
  use 127
property
  create 487
  delete 489
publication date
  set for an object 198

Q

QA staff member 37
QA view 29

R

reference
  absolute 48
  add in Content client 204
  add in editor 209
  add in integrated HTML editor 206
  delete 214
  dependent objects 236
  external 47
  in general 46
  maintain 201
  relative 48
referenced objects
  References dialog box 202
  WCM tag 430
References dialog box
  External references 203
  Invalid references 203
  referenced objects 202
  referencing objects 202
referencing objects
  References dialog box 202
  WCM tag 430
reject
  delete 231
  object 255
rejected (object status) 36
relative references 48
relator 339
  add 341
  add with Livelink document 350
  delete 362
  edit content 354
  edit metadata 355
  integrate metadata 358
  update manually 361
release
  cascading template 405
  delayed 198
  directly 253

object 251
release date
metadata item 45
WCM tag 430
released (object status) 36
released by
metadata item 45
WCM tag 430
remove
attribute from attribute set 483
page 232
reference 214
workflow 284
workflow assignment 284
replace object 190
reset
access rights 226
rights
add principal 216
edit 215
in general 50
set rights for an object 220
role 51
add to access control list 216
set access rights 220
withdraw access rights 222

S

score 101
Script (object type) 69
search 100
objects 365
search functions 96
security warning
Download applet 169
integrated HTML editor 114
workflow modeler 264
select
view 77
website 76
send e-mail 92
server 27
set (data type) 437
set of e-mail addresses (data type) 438
several objects
edit 365
edit metadata 196
source object
copy 236
delete 241
edit 240
metadata 236
move 236
reconvert 235
staging actions 235
submit 241
special attribute
Absolute URL 416
Statify page 415
Timeout 415
special attributes
edit 411
filter by 412
from attribute set 409
from object category 410
in general 409
statification 413
WCM tags 433
staging 29
standard filter 365
application examples 373
in the My objects view 369
in the object filter 368
use 366
standard metadata 43
start
Content client 74
InSite Editing 295
workflow modeler 263

starting point
change size 268
delete 270
move 267
statification 413
of form instances 418
of JSP objects 416
of XML objects 421
Statify page (special attribute) 415
status
metadata item 45
WCM tag 430
status bar 81
object information 86
string (data type) 438
structure
cascading template 402
WCM object 38
workflow definition 258
submit 248
compound object 241
source object 241
undo 250
submitted (object status) 36
substitute 79
suggested file name
metadata item 45
WCM tag 430
surrogate (object data type) 471
system architecture 27

T

tab
External (References dialog box) 203
Invalid (References dialog box) 203
Referenced objects (References dialog box) 202
Referencing objects (References dialog box) 202
take over as substitute 79
target group
metadata item 45
WCM tag 430
tasks
of a QA staff member 37
of an editor 37
template
add 399
add object 138
cascading 401
delete 400
edit 400
in general 41, 397
merge with object 398
metadata item 46
WCM tag 430
template structure 83
text (data type) 439
thumbnails
activate 422
time (data type) 439
time stamp (data type) 439
time-consuming actions 101
timeout (special attribute) 415
title
metadata item 46
WCM tag 430
title bar 81
toolbar
configure 116
Content client 81
workflow modeler 265
topic
object data type 471
object type 69
topic structure 82
transition
add 268

delete 270
move 269
type
metadata item 46
WCM tag 430

U

undo
changes 186
check out 173
submit 250
update
relator manually 361
update an object 189
Update Edit version automatically
folder relator 349
relator 345
URL
WCM tag 431
URL of the statified page (WCM tag) 431
URL of the surrogate page (WCM tag) 431
user
add to access control list 216
set access rights 220
withdraw access rights 222
user interface 81

V

variable in filter 382
version 87
metadata item 46
restore 186
WCM tag 431
view
Edit 29
my objects 97
my worklist 98
object list 84
Production 29
QA 29
search 100
select 77
template structure 83
topic structure 82

W

WCM object 27
access rights 50
content 39
metadata 42
references 46
status 32
structure 38
template 41
type 65
version 87
WCM tag
add (parameter) 461
buttons (parameter) 460
default (parameter) 452
edit (parameter) 458
encoding (parameter) 450
filter (parameter) 451
for Livelink standard metadata 358
for user-defined Livelink metadata 359
format (parameter) 441
in general 427
insert in HTML editor 179
integrate global information 431
integrate metadata 428
integrate object content 428
integrate special attributes 433
integrate standard metadata 428
integrate workflow information 432
label (parameter) 453
locale (parameter) 449
oid (parameter) 457
option_id (parameter) 454

option_mode (parameter) 455
order (parameter) 459
parameters 440
required (parameter) 453
separator (parameter) 448
target (parameter) 461
thumbnails (parameter) 452
type (parameter) 450
WCM tag (insert)
enable InSite Editing (parameter) 183
format (parameter) 182
label (parameter) 182
locale (parameter) 182
OID (parameter) 182
position in the Metadata dialog (parameter) 184
required (parameter) 182
select metadata item 179
specify parameters 181, 183
starting wizard 179
type (parameter) 182
WCM tag dialog box 299
web document (object data type) 471
website
manage 465
wizard
for adding a relator 341
for adding an object based on a template 148
for adding objects based on object type 134
for editing metadata of several objects 196
Word document (object type) 69
workflow 257
application example 287
assign 282
create 272
delete 280
display information 92
edit 278
edit object 285
functional area 62
introduction 30
object data type 472
object type 69
remove assignment 284
workflow assignment
create 282
remove 284
workflow definition
create 272
delete 280
edit 278
structure 258
workflow elements
delete 270
graphical representation 270
workflow modeler
functions 265
in general 262
install 261
menu bar 265
start 263
toolbar 265
World 52
add zu access control list 216
set access rights 220
withdraw access rights 222

X

XML
application examples 500
object types 493
use templates 498
XML document
object data type 472
object type 69
XML schema
object type 70

XML template
    object data type 472
    object type 70
XSLT document (object type) 70
XSLT template
    object data type 472
    object type 70